

THESES

présentées devant l'UNIVERSITE de NANTES

pour obtenir le grade de

DOCTEUR-INGENIEUR

par

Michel AZEMA

ECOLE NATIONALE SUPERIEURE DE MECANIQUE



1^{ère} Thèse : Commande sous optimale par algorithmes d'apprentissage
Application à la commande en ligne d'une chaudière.

2^{ème} Thèse : Propositions données par le Jury

Soutenue le 12 septembre 1970 devant la Commission d'Examen

M. MEZENCEV Président

MM. LE CORRE	} Examineurs
THOMAS	
de LARMINAT	
CARMASSI	

THESES

présentées devant l'UNIVERSITE de NANTES

pour obtenir le grade de

DOCTEUR-INGENIEUR

par

Michel AZEMA

ECOLE NATIONALE SUPERIEURE DE MECANIQUE



1^{ère} Thèse : Commande sous optimale par algorithmes d'apprentissage
Application à la commande en ligne d'une chaudière.

2^{ème} Thèse : Propositions données par le Jury

Soutenue le 12 septembre 1970 devant la Commission d'Examen

M. MEZENCEV Président

MM. LE CORRE	} Examineurs
THOMAS	
de LARMINAT	
CARMASSI	

232311

DOCTEUR-INGENIEUR

ANNA HENRI



UNIVERSITE DE NANTES

Président M. KERNEIS Médecine

Vice-Président M. PETRESCO Mathématiques

• • • • •

LISTE DES ENSEIGNANTS DES DISCIPLINES SCIENTIFIQUES

Professeurs

M. BRILLOUET

Mathématiques

M. MOLINARO

"

M. PAUC

"

M. PETRESCO

"

M. RAFFIN

"

M. VIVIER (E.N.S.M.)

"

M. TARDY (E.N.S.M.)

Mécanique

M. RAVILLY (E.N.S.M.)

"

M. AVIGNON

Physique

M. BOULLLOUD

"

M. COLLET

"

M. MINN

"

M. MEZENECY (E.N.S.M.)

"

M. MOUTON

"

M. DUNOYER

Chimie

M. GRAFF

"

M. PIERI

"

M. QUINIQU

"

M. WOJTKOWIAK

"

Mlle DEBRAUX

Biologie

M. GARRIGUES

"

Mme LE DOUARIN

"

M. LE DOUARIN

"

M. SELLIER

"

M. CAVET

Sciences de la Terre

Cette étude a été menée au Laboratoire d'Automatique de l'ENSM à Nantes. J'adresse tous mes remerciements à Monsieur le Professeur Roman Nezeux, Directeur du Laboratoire, qui m'a accueilli au sein de son équipe de chercheurs et ne m'a jamais privé de son soutien.

Monsieur Jean-François Le Corre, Maître de Conférences à l'IST de Nantes, me fait l'honneur de participer à mon jury, je lui en suis très reconnaissant.

Monsieur Yves Thomas m'a dirigé dans mon travail et m'a offert l'appui de ses connaissances, je lui exprime ma profonde gratitude.

Je remercie Monsieur Philippe De Laminat, qui m'a prodigué son savoir et ses conseils.

Je suis très reconnaissant envers Monsieur Michel Cammari, ingénieur de Développement à la SNPA, d'avoir bien voulu considérer mon travail avec bienveillance et participer à mon jury.

Je remercie également l'Equipe du Centre de Calcul de l'ENSM-IST et son chef, Monsieur Chèvre, pour leur aide précieuse.

J'adresse enfin mes remerciements à tous les chercheurs du Laboratoire d'Automatique, ainsi qu'aux techniciens et en particulier Messieurs Bané, Bellanin et Rougé, dont l'habileté et la conscience professionnelles ne sont pas les moindres qualités.

Nicolas Ming



Digitized by the Internet Archive
in 2026

https://archive.org/details/IA41555016_0024

TABLE DES MATIERES

INTRODUCTION

CHAPITRE I ETUDE DU PROCESSUS PILOTE

I - 1 Validité du processus pilote

- I-1-1 Généralités sur le fonctionnement des Centrales thermiques
- I-1-2 Principe de fonctionnement d'un Générateur de vapeur industriel
- I-1-3 Présentation et caractéristiques générales de l'Unité pilote
- I-1-4 Comparaison entre Unité industrielle et Unité pilote
- I-1-5 Le problème principal de régulation

I - 2 Description des régulations annexes du processus pilote

- I-2-1 Régulation de niveau d'eau dans le ballon
- I-2-2 Réglage de la combustion A - Réglage de l'excès d'air
 - 1- Mesure indirecte de -'excès d'air
 - 2- Analyse des fumées
 - 3- Détermination de la fonction $E = f(q_c)$
 - 4- Régulation du débit d'air
 - 5- Mesure de débit d'air comburant
- B - Réglage de la dépression
 - 1- Généralités
 - 2- Régulation

I - 3 Identification du processus pilote

- I-3-1 Méthode
- I-3-2 Etude détaillée
- I-3-3 Résultat

I - 4 Conclusion

CHAPITRE II LA COMMANDE SOUS OPTIMALE - THEORIE

II - 1 Caractérisation de h

- II-1-1 Reconnaissance de forme et systèmes à apprentissage
- II-1-2 L'élément linéaire à seuil
 - A- Description
 - B- Choix du type d'entrées
 - C- Codage des variables d'entrée
- II-1-3 Propriétés de l'élément linéaire

II-1-4 Choix d'un code

II-1-5 Nouvelle caractérisation de l'élément linéaire généralisé

II - 2 Cas où le nombre de catégories est supérieur à deux

II - 3 Mise en oeuvre du générateur de fonction adaptatif

II-3-1 Choix et présentation des échantillons

II-3-2 Méthode d'apprentissage - Les Algorithmes

1- Algorithme à incrément fixe

2- Algorithme par relaxation

3- Algorithme à β variable

II - 4 Exemple d'application

CHAPITRE III APPLICATIONS DE LA COMMANDE SOUS OPTIMALE

III - 1 Etude en simulation de la commande d'un processus en $1/s^2$

III-1-1 Position du problème d'optimisation

III 1-2 Choix d'un échantillon de solutions exactes

III-1-3 Quantification - Choix d'un code

A- Quantification de la commande

B- Quantification de l'état

C- Choix d'un code

III-1-4 Apprentissage

III-1-5 Simulation de la commande sous optimale

A- Mise en oeuvre du calculateur analogique

B- Mise en oeuvre du calculateur numérique

C- Mise en route

III-1-6 Résultats expérimentaux et conclusion

III - 2 Application de la méthode d'apprentissage à un problème réel

III-2-1 Position du problème

III-2-2 Calcul des trajectoires optimales

III-2-3 Essai en simulation de la commande optimale

A- Calcul des équations "machine"

B- Câblage

C- Résultats

III-2-4 Apprentissage

A- Quantification - Choix des échantillons

B- Le calcul hors ligne - Algorithmes

C- Résultats

III-2-5 Essai en simulation de la commande sous optimale

A- But de cet essai

B- Câblage

C- Résultats

III-2-6 Application au processus pilote

A- Mise en oeuvre

B- Résultats

CONCLUSION

ANNEXE

BIBLIOGRAPHIE

- INTRODUCTION -

Avant tout, précisons que le terme "optimal" doit être pris ici au sens des systèmes dynamiques décrits par des équations différentielles ordinaires de la forme $\dot{\underline{x}} = \underline{f}(\underline{x}, \underline{u}, t)$ où $\underline{x}$ est le vecteur d'état et $\underline{u}$ le vecteur de commande. La commande est dite optimale si elle minimise une fonctionnelle:

$$J(\underline{u}) = \int_{t_0}^{t_0+T} f_0(\underline{x}, \underline{u}, t) dt$$

sachant que $\phi(\underline{x}(t_0), t_0) = 0$ et $\psi(\underline{x}(t_0+T), t_0+T) = 0$, et compte tenu du fait que des contraintes égalités et inégalités telles que $\underline{u} \in U$ (contraintes sur la commande), $\underline{x} \in X$ (contraintes sur l'état), U et X représentant des domaines fermés, ou encore $\partial(\underline{x}, \underline{u}) \leq 0$ (contraintes mixtes), sont imposées sur le processus, soit naturellement, soit par l'utilisateur. Généralement, l'état du système n'est pas directement mesurable mais déterminé à partir d'un vecteur d'observation $\underline{y} = \underline{h}(\underline{x}, \underline{u}, t)$.

La résolution de ce type de problème est actuellement assez bien développée et fait l'objet des deux théories sans doute les plus importantes de l'automatique :

- les méthodes variationnelles, qui fournissent des conditions nécessaires d'optimalité sous la forme d'un système différentiel dont les conditions aux limites sont imposées en plusieurs points.
- les méthodes d'Invariant Imbedding (Programmation Dynamique) qui fournissent des conditions d'optimalité globale sous la forme de système de Cauchy aux dérivées partielles.

Les besoins militaires et aérospatiaux d'une part, l'avènement des calculateurs digitaux et hybrides, puissants et rapides, d'autre part, ont permis un développement extraordinaire de l'optimisation depuis maintenant une douzaine d'années.

Parmi les applications les plus fréquentes on relève:

- le changement d'orbite avec minimalisation du combustible
- les rendez-vous
- le guidage des fusées interplanétaires en minimalisant le combustible et l'écart par rapport au but assigné
- le problème du retour dans l'atmosphère
- l'atterrissage sur la Lune
- les missiles sol-air
- le contrôle d'attitude des satellites.

Outre leur grande motivation, ces différents problèmes sont parfaitement posés en ce sens que le critère d'optimalité est très bien défini (Le plus souvent : consommation minimale avec but précis à atteindre) et que le processus (Vaisseau spatial, fusée) a un comportement dynamique connu avec précision.

De là à étendre ces puissantes théories à d'autres domaines et en particulier celui de la régulation industrielle (Réacteurs chimiques, colonnes à distiller, fours à ciment, procédés de cracking, centrales thermiques ...), il n'y avait bien entendu qu'un pas qui, si l'on en juge par la quantité des publications relativement faible, est encore aujourd'hui incomplètement franchi.

Pourtant une importante motivation existe: Les théories classiques des servomécanismes et de la régulation sont à peu près inopérantes en présence de systèmes multivariables, non stationnaires, non linéaires et en présence de contraintes.

Mais des obstacles non négligeables sont apparus :

- l'Industriel est généralement assez satisfait de ses procédés de régulation classiques (PID, calculateurs analogiques), qu'il connaît bien.
- la mise en équations du processus à réguler (ou détermination du modèle) est une opération complexe et souvent longue et onéreuse. Or, cette étape est fondamentale pour l'optimalisation.
- il est pratiquement impossible de déterminer un critère de performance ($J(u)$) de forme simple et susceptible d'être "mesuré" à court terme. Les buts d'une régulation industrielle sont en effet très nombreux et doivent être pondérés d'une manière essentiellement subjective. Le critère n'est en fait constitué que de tendances de bon fonctionnement.
- la résolution en ligne des conditions d'optimalité fait appel à de puissants calculateurs et à des programmes sophistiqués de spécialistes, et ne fournit dans le cas général que des commandes en boucle ouverte.

Ceci nous amène à penser que, sauf cas particuliers de nécessité de commander un processus dans un but très précis, les théories de l'optimalisation ne doivent constituer que des indications, des informations pour mettre en oeuvre une meilleure commande que les régulations classiques. D'où le terme de " sous-optimalité " utilisé dans le titre de notre rapport.

Il y a bien entendu différentes manières d'envisager la sous-optimalité, pour laquelle nous avons retenu les principes suivants :

- 1- Dans la mesure où on désire un bon fonctionnement, cela implique que l'on s'attache bien davantage à des allures de réponses plutôt qu'à la valeur numérique d'un critère de performance qui, cependant, de par son caractère global, sera utilisé pour le calcul des commandes. De plus, une certaine marge dans la forme de réponse est permise et ainsi il est peu important d'avoir une grande précision dans les mesures. Leur localisation grossière est suffisante. Ce point de vue nous amène à quantifier les mesures, ce qui permettra une structure simple de commande.
- 2- Cette simplicité ne doit pas empêcher la commande d'être supervisable, dans un but d'adaptation, par un calculateur numérique .
- 3- La commande doit s'effectuer en boucle fermée car d'une part on réduit normalement la sensibilité des performances aux variations et aux inexactitudes des paramètres du processus, et d'autre part ce type de commande est particulièrement efficient en présence de perturbations toujours existantes pour lesquelles on ne peut pas même définir de caractéristiques statistiques.

Nous précisons maintenant sur un exemple les différentes étapes de la méthode utilisée :

Supposons qu'on ait un système d'équation $\dot{x} = u$, avec une commande u telle que $u(t) \leq B = \text{cte.}$ et qu'on ait pu convertir les caractéristiques requises sur la réponse en la minimisation d'un critère tel que :

$$J(u) = \frac{1}{2} \int_t^{t+T} (x^2 + \alpha^2 u^2) dt \quad T \text{ fixé}$$

On détermine tout d'abord à l'aide d'algorithmes classiques d'op-

timalisation quelques solutions exactes à partir desquelles une fonction explicite $u = h(x(t), B, \alpha, T) = h(\underline{p})$ doit être obtenue .

La structure de h est prise de la forme $\underline{W}^T \underline{P}_1$, où $\underline{P}_1$ est le vecteur $\underline{p}$ ci-dessus , mais quantifié .

Le vecteur $\underline{W}$ est déterminé au moyen d'algorithmes utilisés en reconnaissance de forme. La fonction h est alors appliquée en ligne sur le processus.

Cette méthode a été mise au point sur ordinateur hybride et testée sur l'ensemble évaporateur-surchauffeur d'un générateur de vapeur pilote que nous décrivons dans une première partie en insistant particulièrement sur ses similitudes avec un générateur de Centrale thermique.

La deuxième partie est ensuite consacrée à l'exposé théorique de la méthode sous-optimale , tandis que le dernier chapitre contient de multiples résultats expérimentaux concernant un problème simulé avec contraintes sur l'état et la commande du processus réel.

ETUDE DU PROCESSUS PILOTE

A partir d'une description succincte des problèmes de régulation inhérents aux différents types de centrale thermique existant en France, nous étudions dans ce chapitre la validité et les limites de l'unité pilote dont nous disposons. Ceci nous amène à distinguer des problèmes de régulation annexes et principaux pour l'unité pilote. Nous exposons ensuite une méthode d'obtention d'un modèle mathématique du processus. Enfin, nous posons les équations de ce modèle, tel qu'il sera utilisé dans la suite de cette étude.

I - 1 VALIDITE DU PROCESSUS PILOTE

I - 1-1 Généralités sur le fonctionnement des centrales thermiques

A partir d'informations provenant du dispatching général installé à PARIS, et que l'on peut schématiser suivant la figure 1, chaque dispatching régional prévoit la charge pour chaque Centrale, pour une période de 24 heures. A ces programmes de charge, prévus par des données statistiques, s'ajoutent des perturbations relativement importantes, considérées à l'heure actuelle comme aléatoires. Ces perturbations peuvent avoir comme origine des accidents mécaniques, des conditions atmosphériques défavorables, des événements politiques ou sportifs, etc... Leur amplitude est cependant réduite, grâce à la participation instantanée du réseau Européen au recouvrement de l'incident, le pays responsable restituant ensuite à l'étranger la puissance précédemment empruntée.

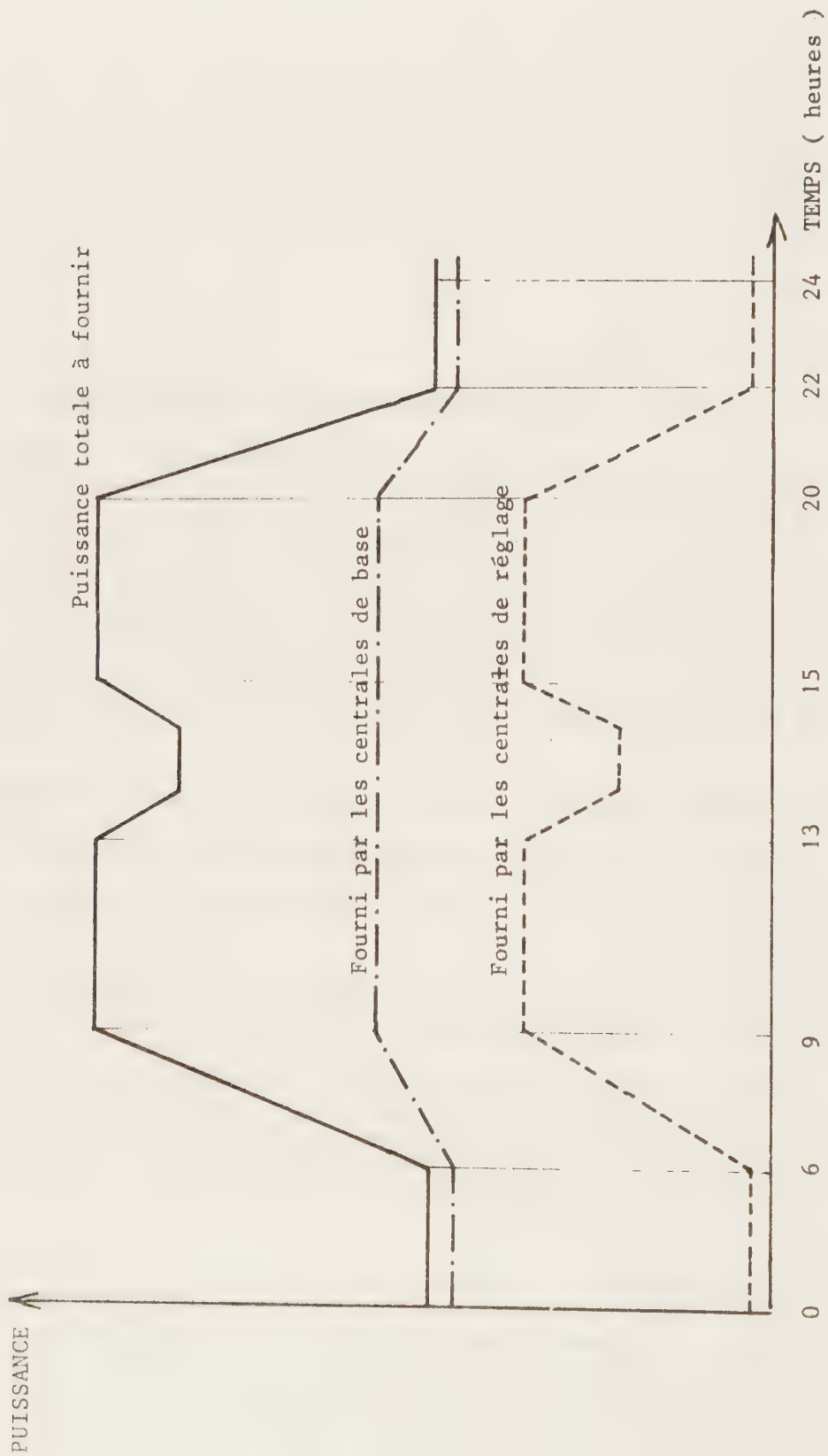
Dans une première approche, on peut distinguer deux types de centrales:

- Les centrales de base (très importantes centrales thermiques, centrales nucléaires)
- Les centrales de télé-réglage (la plupart des centrales thermiques de moyenne puissance)

Le rôle des centrales de base est de fournir un certain volume de puissance. Leurs variations de charge sont lentes et de faible amplitude.

Les centrales de télé-réglage fournissent le complément de puissance et participent également au réglage de la fréquence. Elles fonctionnent à puissance essentiellement variable, aussi bien sur une période de 24 heures comme le montre la figure 2, où sont représentées les variations de puissances d'une centrale de télé-réglage (NANTES-CHEVIRE, 19 Décembre 1969), qu'au cours d'une année.

Ces centrales présentent un grand intérêt pour nous : les perturbations auxquelles elles doivent faire face font que se posent des problèmes de régulation difficiles à résoudre. En effet, quelle que soit la charge électrique de la centrale, autrement dit quelle que soit la demande de vapeur, la pression et la température de cette vapeur doivent rester entre certaines limites très étroites ($\pm 5\%$ en pression, $\pm 6^\circ\text{C}$ en température pour une vapeur à 160 bars et 550°C). Or si le problème de la pression, en lui-même ne présente pas de difficulté majeure, il n'en est pas de même pour la température dont les réglages actuels ne donnent pas satisfaction (du fait, en particulier, du couplage important pression-température).



Shéma de la variation de la puissance sur le réseau Français, pour une période type de 24 heures.

- FIGURE 1 -

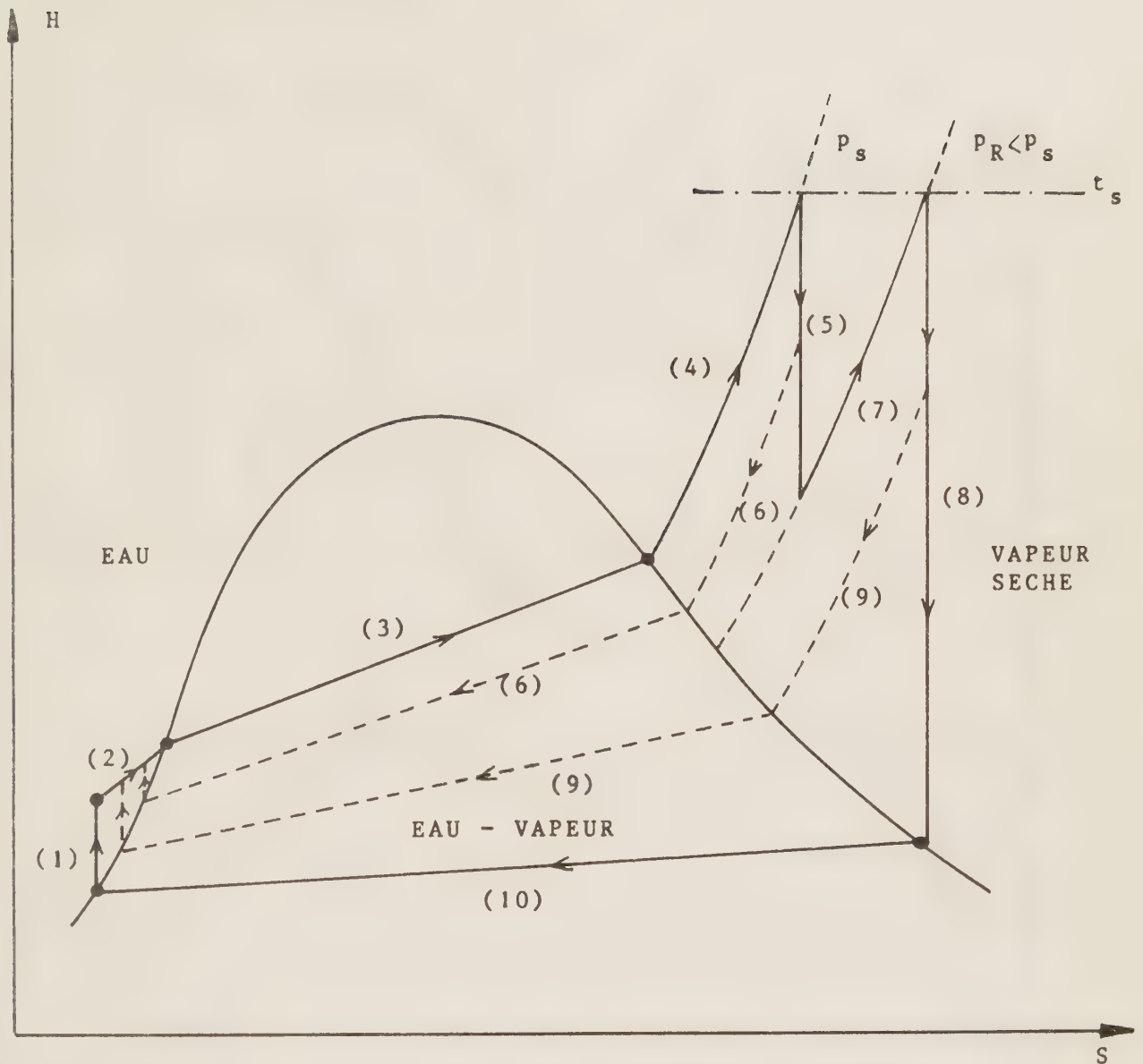
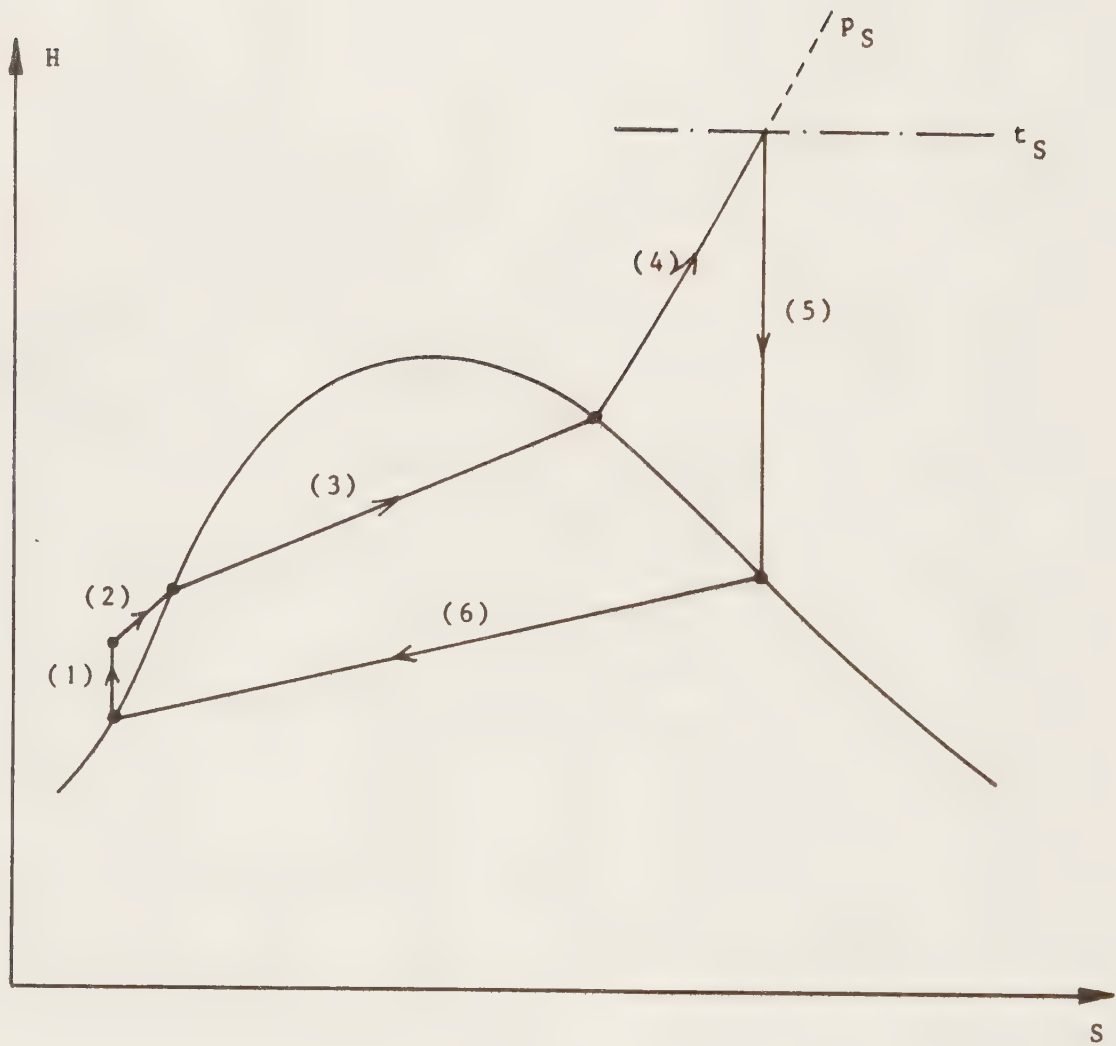
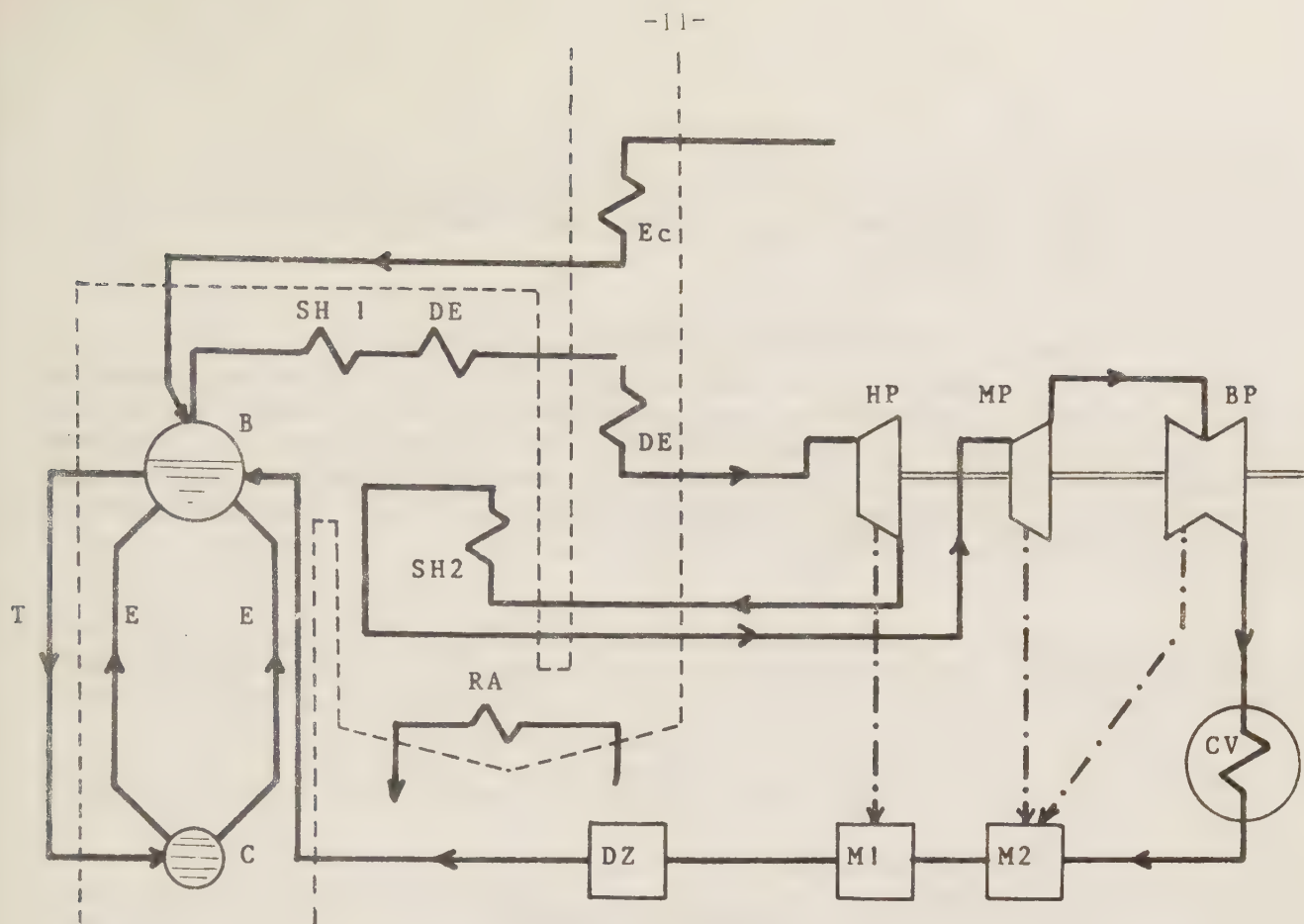


Diagramme de MOLLIER - Cycle à surchauffe, resurchauffe isotherme et soutirages. (théorique)



Cycle à surchauffe. (théorique)



Circuit eau-vapeur d'une chaudière industrielle. (principe)

Description :

B	Ballon	MP	Turbine moyenne pression
C	Collecteur inférieur	BP	Turbine basse pression
T	Tube de descente	CV	Condenseur sous vide
E	Ecran d'eau	M1	Mélangeur HP (vapeur soutirée)
SH1	Surchauffeurs	M2	Mélangeurs MP et BP
SH2	Resurchauffeurs	DZ	Dégazeur
DE	Reglage de la surchauffe	RA	Réchauffeur d'air
HP	Turbine haute pression	Ec	Economiseurs

Comme précédemment, nous donnons une description succincte du circuit eau-vapeur. (figure 6)

L'eau, préalablement rechauffée dans les tubes de descente, par la circulation des fumées, est vaporisée dans les écrans. La vapeur est recueillie dans le ballon puis surchauffée. Le surchauffeur est scindé en deux groupes symétriques, le réglage fin de la température étant effectué entre les deux groupes par injection d'eau froide. La vapeur est ensuite envoyée sur une turbine à un seul étage (B.P.) puis est condensée à l'atmosphère et réinjectée dans le circuit de chauffe après dégazage.

Le cycle eau-vapeur traduit ces différentes phases (figure 5), on peut remarquer la simplification par rapport à une installation industrielle. Cette simplification est également apparente sur le schéma de l'installation figure 6.

Le cycle est décrit de la manière suivante : (figure 5)

- 1 - Pompage de l'eau
- 2 - Réchauffage
- 3 - Vaporisation (isotherme et isobare)
- 4 - Surchauffe (isobare)
- 5 - Détente en turbine (isentropique)
- 6 - Condensation

I - 1 - 4 Comparaison entre unité industrielle et unité pilote

Cette comparaison fait ressortir trois caractéristiques principales :

A - La chaudière pilote elle-même possède les caractéristiques essentielles du circuit vapeur de centrales :

- Cycle à surchauffe
- Grandeurs réglantes principales : débit de combustibles, injection d'eau de désurchauffe.

Cette similitude conserve le problème fondamental du réglage optimal des deux grandeurs : pression et température.

B - La charge de l'unité pilote n'a aucune commune mesure avec celle d'une centrale (en particulier, la turbine n'a qu'un seul étage, et le réseau électrique n'existe pas).

Cependant, dans une centrale, le débit de vapeur d'admission à la turbine suit sensiblement les variations de la puissance demandées à la centrale. Du point de vue de l'automatique, on peut donc admettre la charge de la chaudière pilote comme partie non-intégrante de l'ensemble à commander et considérer le débit de vapeur comme une variable indépendante dont les caractéristiques déterministes et statistiques sont celles de la puissance demandée à une centrale.

C - Le circuit d'économiseurs est inexistant sur le pilote.

Cette caractéristique réduit d'une part le nombre de régulations annexes à mettre en place, mais d'autre part crée des phénomènes parasites dans le ballon, dus à une injection d'eau trop froide. Ce dernier point complique inutilement le problème de régulation.

Nous pouvons admettre, en conclusion, que l'intérêt présenté par notre installation pilote est de pouvoir cerner le problème du réglage de la pression et de la température quelles que soient les conditions de charge, sans se préoccuper outre mesure d'impératifs techniques souvent complexes et finalement annexes.

I - 1 - 5 Le problème principal de régulation

Nous avons vu que les perturbations aléatoires modifiaient sensiblement l'allure prévisible de la courbe de charge d'une unité thermique. Toutes les régulations étant assurées par ailleurs, ces perturbations ont un effet direct sur la pression et la température de la vapeur surchauffée, dont le couplage est important du fait que ces grandeurs sont toutes deux influencées par le Flux calorifique.

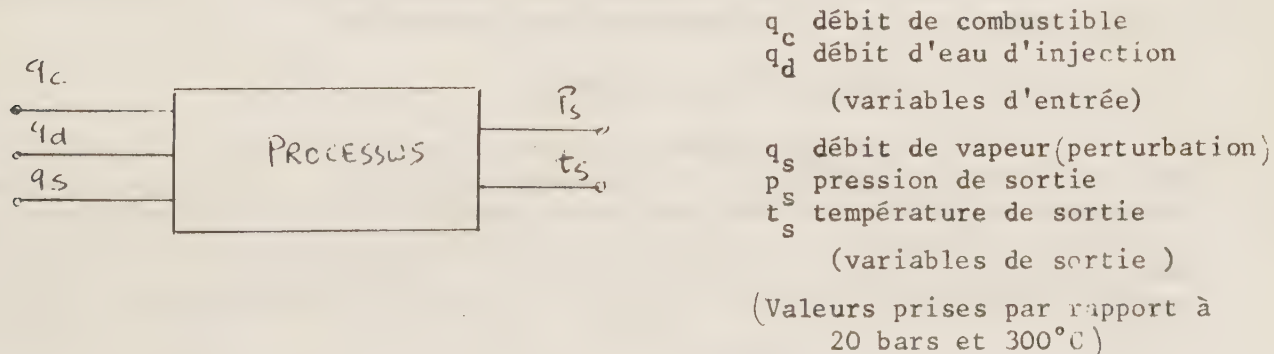
Du point de vue des perturbations, en ce qui concerne notre pilote, nous devons donc régler température et pression de vapeur selon certains critères de fonctionnement :

- Soit en présence de rampes de charge
- Soit lors de charge moyenne constante soumise à des perturbations aléatoires.

Cette régulation doit avoir pour caractéristique principale la rapidité de réponse. Mais elle doit également tenir compte des contraintes physiques auxquelles peut être soumis le processus : citons par exemple le gradient de température sur les tubes de vaporisation (corps de feu) le gradient de pression sur les ailettes de la turbines etc...

D'autre part, et indépendamment des contraintes, il existe des tolérances à respecter sur les écarts de pression et de température, par exemple $\pm 5\%$ sur la pression et $\pm 5^\circ\text{C}$ sur la température, dans le cas d'une centrale, ce qui revient à ± 1 bar à $\pm 5^\circ\text{C}$ dans le cas de notre pilote.

En résumé, l'unité pilote peut être schématisée en tenant compte uniquement des grandeurs réglantes et réglées principales (p et t d'une part, q_c q_d et q_s d'autre part), les autres grandeurs n'apparaissant pas explicitement.



Ce schéma a l'avantage d'englober les régulations annexes bien que les grandeurs réglées correspondantes ne figurent pas explicitement en tant que entrées ou sorties.

I - 2 Description des régulations annexes du processus pilote

Les régulations annexes installées sur le pilotes sont considérées comme annexes en raison soit de leur action rapide soit de leur faible couplage avec les grandeurs principales.

Ce sont :

- La commande d'eau alimentaire (régulation de niveau)

Remarquons que le problème de la régulation de niveau au ballon de notre pilote est moins délicat à résoudre que dans une unité industrielle, où les dimensions du ballon ne permettent pas de variations de niveau importantes.

- Le réglage de la combustion (commande du débit d'air et de la dépression au foyer).

I - 2 - 1 Régulation de niveau d'eau dans le ballon

Le niveau d'eau dans le ballon doit être maintenu constant quelle que soit la demande de vapeur. Nous avons à notre disposition deux systèmes de réglage :

- Un réglage mécanique
- Un réglage électronique à servovanne

Nous avons dans ce cas adopté un réglage par action proportionnelle et par tendance :

Si δq_{EA} est la commande appliquée au servo-moteur, on doit avoir :

$$\delta q_{EA} = k_{N1} (N_c - N) + k_{N2} (q_V - q_{EA})$$

k_{N1}	constante de proportionnel	N	Mesure de niveau
k_{N2}	constante de tendance	q_V	Débit de vapeur au ballon
N_c	consigne de niveau	q_{EA}	débit d'eau alimentaire

La commutation de l'un à l'autre système se fait par action sur un relais pneumatique relié à deux vannes complémentaires.

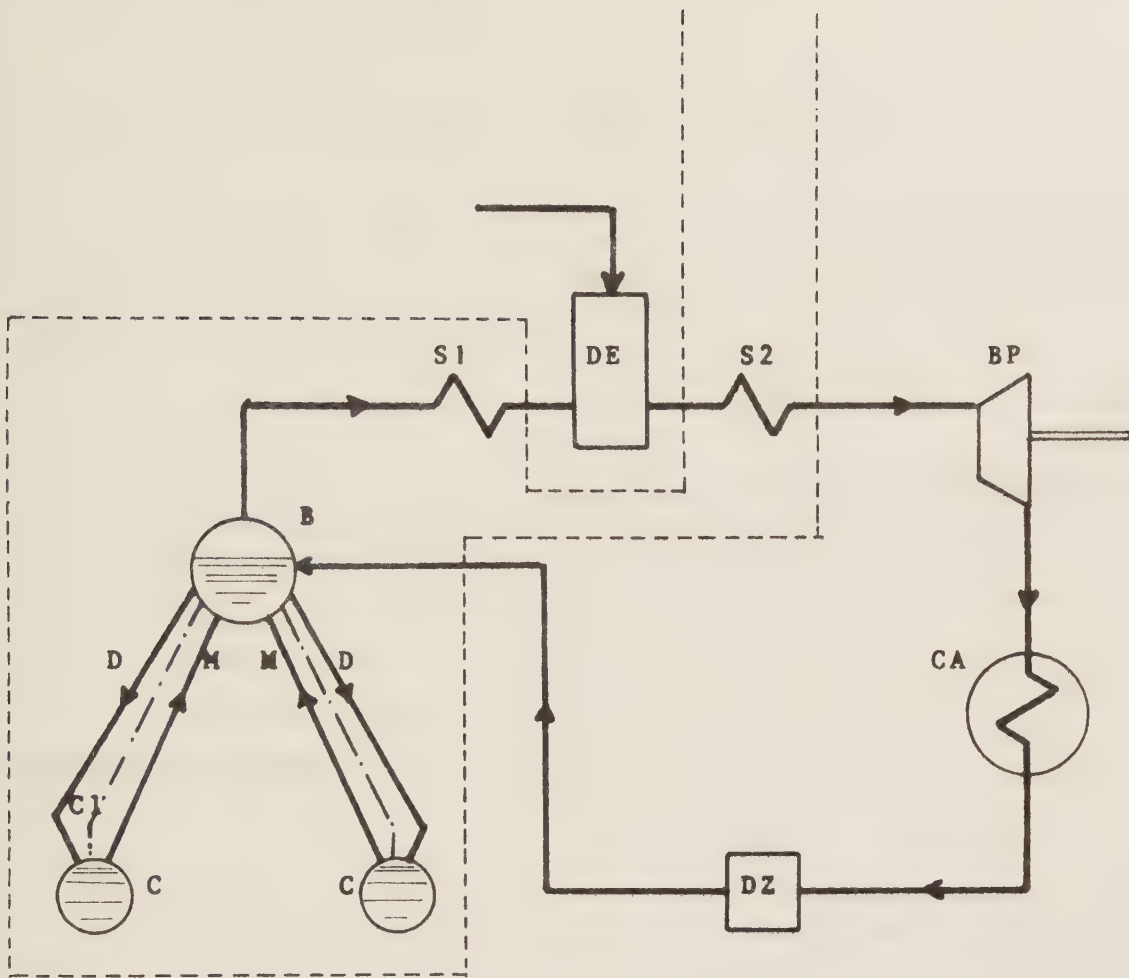
I - 2 - 2 Réglage de la combustion

Pour obtenir le rendement optimal d'une combustion il importe d'une part de limiter les pertes par la cheminée (trop d'air), d'autre part de réduire au minimum les pertes par imbrûlés gazeux (manque d'air). Ces considérations impliquent la détermination de l'excès d'air dans les gaz à la sortie du foyer (2), par dosage du gaz carbonique. D'autre part, la combustion ne peut se faire d'une manière satisfaisante que si la chambre de combustion est en légère dépression (tirage).

Le réglage de la combustion nécessite donc la mise en place de deux systèmes de réglage : un régulateur de débit d'air comburant, un régulateur de dépression dans le foyer.

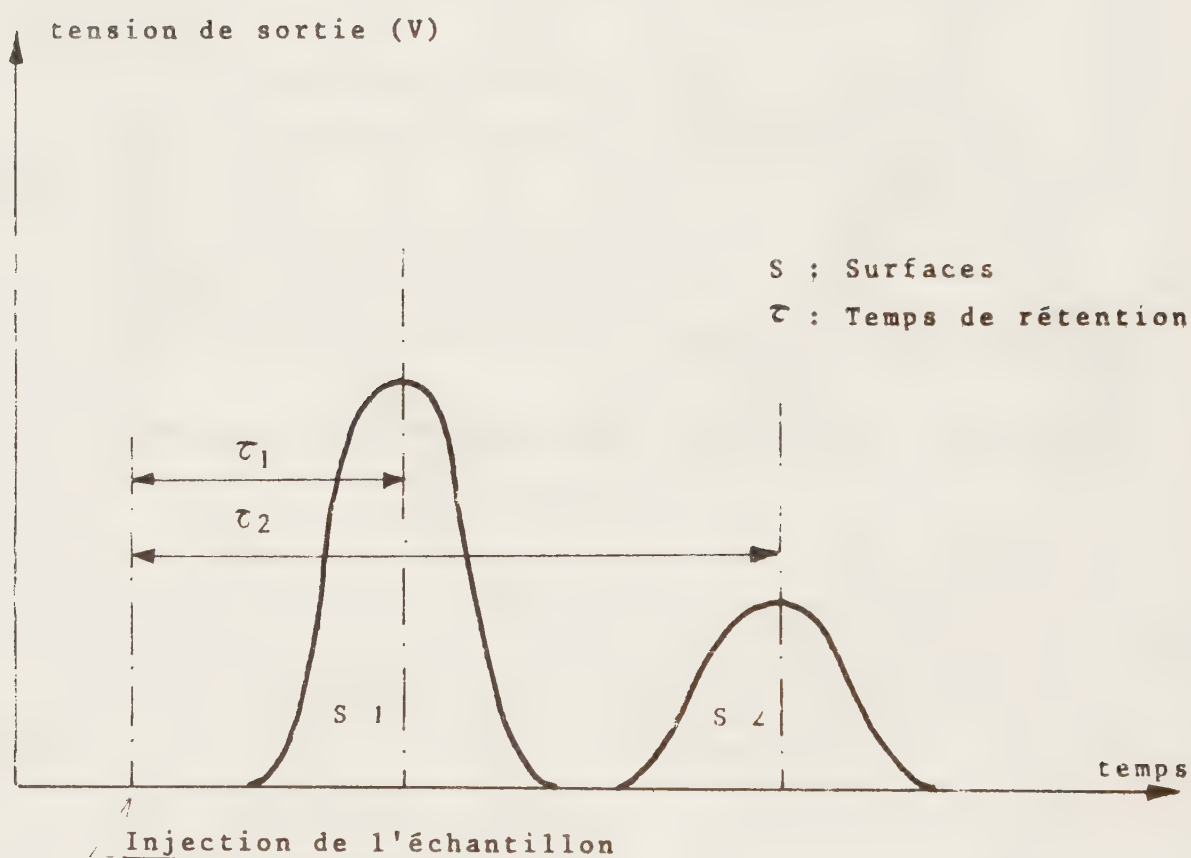
A - Réglage de l'excès d'air (commande du débit d'air comburant)

L'analyse des gaz de combustion permet de déterminer, d'une part le volume d'air non utilisé dans la combustion, d'autre part la valeur optimale de l'excès d'air. On établit alors une relation entre excès d'air et débit com-



Circuit eau-vapeur de la chaudière pilote.

B	Ballon	S1, S2	Surchauffeur
C	Collecteurs inférieurs	DE	Desurchauffeur
D	Tubes de descente	BP	Turbine basse pression
M	Ecran d'eau	CA	Condenseur atmosphérique
C1	Cloisons	DZ	Dégazeur



Chromatographie en phase gazeuse. (principe)

- FIGURE 7 -

bustible; Cette relation donne la consigne q_{Ac} à laquelle on compare la mesure q_{AM} dans une régulation de type classique.

- Mesure indirecte de l'excès d'air par dosage de CO2 dans les gaz de combustion :

Lorsqu'on brûle un seul type de combustible dans lequel la composition en hydrogène ne varie que très peu, la teneur en CO2 est liée au volume d'air introduit dans le foyer. (Ceci n'est plus valable dans le cas d'une combustion mixte - fuel, charbon, gaz, le volume de CO2 dépendant de la teneur relative en C et H2, qui varie avec le combustible considéré).

En ce qui concerne notre unité pilote, le combustible utilisé est le fuel domestique.

A partir de la connaissance de la composition du combustible, on détermine son pouvoir calorifique (quantité d'air nécessaire et suffisante pour assurer une combustion neutre) et les teneurs de CO2 et de O2 en fonction de l'excès d'air.

L'excès d'air est défini, en pourcent, comme le rapport suivant :

$$E = 100 \frac{q_A - \lambda q_c}{\lambda q_c}$$

E Excès d'air (%)
 q_A débit d'air réel

q_c débit de combustible
 λq_c pouvoir calorifique
 = 11,720 m³/Kg fuel

2 - Analyse des fumées

Cette analyse a été faite par chromatographie en phase gazeuse (3).

L'échantillon à analyser est injecté dans le chromatographe où il est mélangé à un gaz vecteur, inerte et sous pression. Ce gaz vecteur entraîne l'échantillon gazeux à travers une colonne de séparation.

Cette colonne permet de séparer les divers composants de l'échantillon. Un détecteur, placé en queue de colonne, signale toute modification physique et chimique du gaz vecteur. Le détecteur n'identifie pas le produit mélangé au gaz vecteur, il signale simplement sa présence et sa concentration. En fait, le produit est identifié par son temps de rétention dans la colonne, lequel dépend de la température, de longueur de la colonne, du débit de gaz vecteur. Ces conditions étant fixées, la mesure du temps de rétention permet d'identifier le produit.

La figure 7 montre le résultat de l'analyse d'un échantillon gazeux à deux composants. Cet enregistrement donne, premièrement la nature des composants (par lecture des temps de rétention et comparaison avec un tableau pré-établi), deuxièmement les concentrations :

$$C_1 = \frac{S_1}{S_1 + S_2} \quad (\% \text{ de 1 dans le mélange})$$

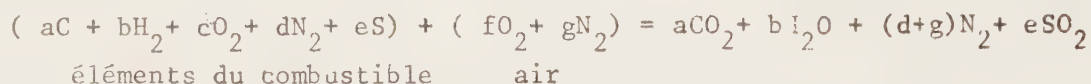
$$C_2 = \frac{S_2}{S_1 + S_2} \quad (\% \text{ de 2 dans le mélange})$$

Dans la réalité, les fumées de chaudière consommant du fuel contiennent : n_2 , O_2 , CO , CO_2 , H_2O , des hydrocarbures légers de cracking et des imbrûlés.

3 - Détermination de la fonction $E = f(q_c)$

Cette détermination se fait de la manière suivante:

On établit une carte de la chambre de combustion, en effectuant des prélèvements d'échantillons. L'analyse de ces échantillons donne un résultat qui est comparé au résultat donné par le calcul théorique à partir de l'équation chimique:



Le point de prélèvement qui donne les résultats les plus voisins est retenu pour la suite.

On effectue ensuite des mesures en ce point de prélèvement pour différents régimes de vaporisation. Pour un débit de combustible donné, on considère comme optimal l'excès d'air qui donne la quantité d'imbrûlés minimum et un débit de vapeur maximum.

Ces mesures permettent d'établir une carte " d'excès d'air conseillé " fonction du débit de combustible (figure 8).

4 - Régulation du débit d'air

Appelons la définition du pourcentage d'excès d'air : $E = \frac{q_A - \lambda q_c}{\lambda q_c}$

Autrement dit, E étant imposé, nous avons une relation entre q_A et q_c :

$$q_A = \lambda q_c (E + 1)$$

Si maintenant, E est une fonction de q_c (cf.3), telle qu'elle a été défini figure 8, nous pouvons écrire :

$$q_{Ac} = f(q_c) + q_{Ao}$$

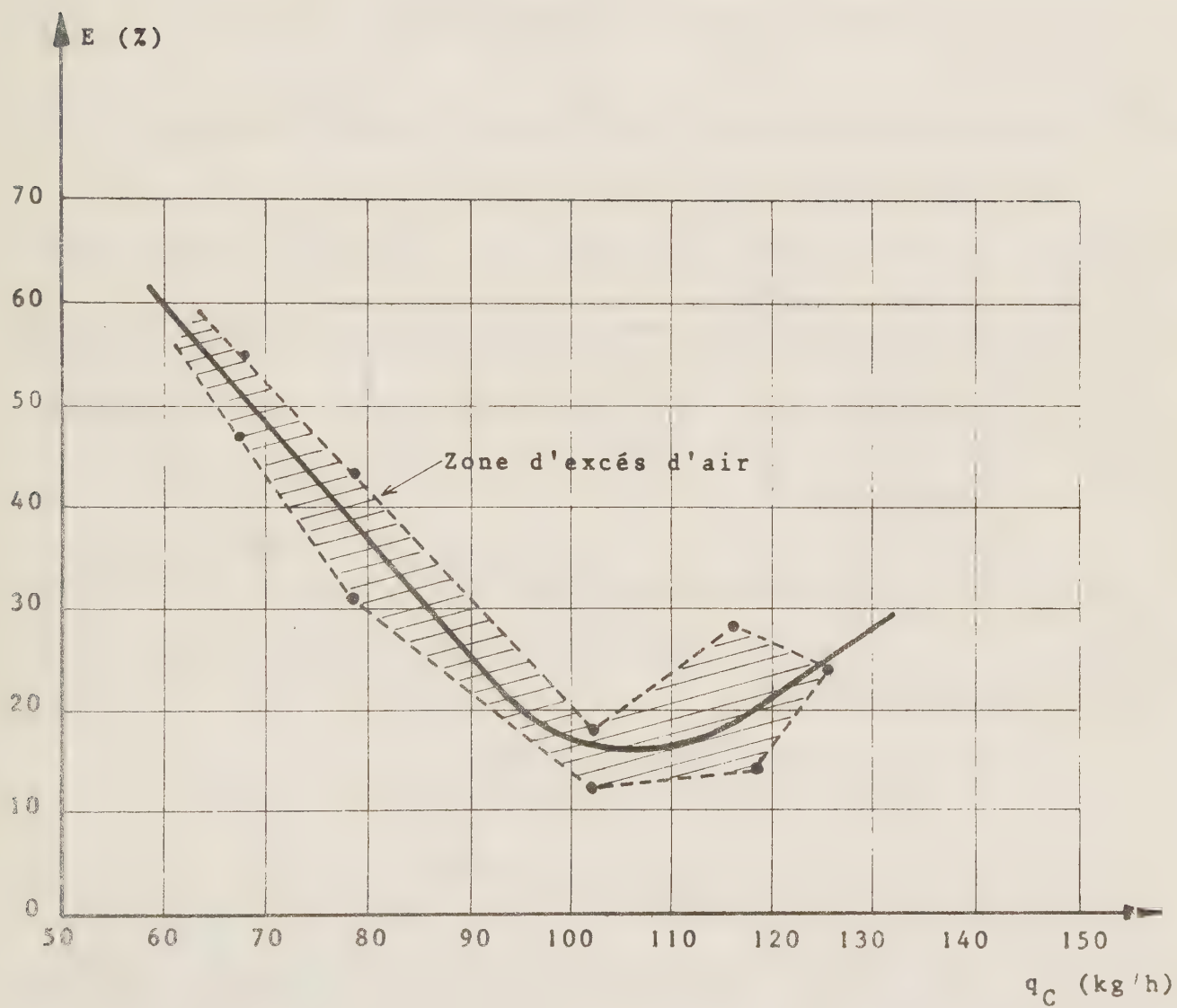
q_{Ac} : débit d'air de consigne (0-10volts)

q_c : grandeur électrique, résultat d'une mesure

q_{Ao} : constante fixée à 3,75 volts

La fonction f est réalisée à partir d'un générateur à diodes.

La chaîne de régulation élabore un écart δq_A entre la référence q_{Ac} et la mesure q_{Am} (cf.5). Cet écart est amplifié (amplificateur magnétique CSF) et appliqué à la phase de contrôle d'un moteur diphasé entraînant une ventelle placée dans le conduit d'amenée d'air.



ZONE d'excès d'air optimale . (déterminée par chromatographie)

- FIGURE 8 -

5 - Mesure de débit d'air comburant

La mesure de débit d'air se fait par l'intermédiaire d'un diaphragme calibré placé dans la canalisation d'amenée d'air. Le débit est proportionnel à la racine carrée de la perte de charge de part et d'autre de ce diaphragme :

$$q_A = k \sqrt{\Delta p}$$

Le coefficient k dépendant (en particulier) des conditions de température et de pression ambiantes, l'imprécision de mesure sur q_A a pu être déterminée à 13%.

B - Réglage de la dépression dans le foyer (commande du tirage)

1 - Généralités

Lorsque la chambre de combustion est en légère dépression par rapport à l'atmosphère, la valeur de cette dépression varie avec l'allure de chauffe (que le tirage soit naturel ou forcé).

Une forte dépression, comme il s'en produit aux allures réduites lorsque on n'agit pas sur le tirage, a pour conséquence des entrées d'air parasites qui ont pour résultat un avilissement du rendement.

Le réglage correct de cette dépression se situe entre 2 et 5mm de C.E. quelque soit l'allure de chauffe.

2 Régulation

Pour un excès d'air donné, et de manière à éviter l'installation d'un capteur de dépression, on peut asservir la position du registre de cheminée au débit de combustible.

Pour cela, on détermine expérimentalement en fonction du débit de combustible, la position du registre de cheminée nécessaire pour maintenir une dépression de l'ordre de -3mm C.E.

On obtient alors la fonction suivante :

$$p_{rD} = f(q_c) + 0,25 \text{ (volts)}$$

p_{rD} : Position du registre de consigne
 q_c : Débit de combustible

Cette consigne est comparée à la position mesurée (p_{rM}). L'écart est envoyé sur un système à relais qui commande un moteur asynchrone (commande 3 positions : Avant - Stop - Arrière). Ce moteur entraîne un système de vis-manivelle couplé au registre.

L'ensemble comporte également des butées de fin de course et un système de commande manuelle par boutons-poussoirs. Enfin une dernière possibilité existe de commander manuellement le registre de cheminée à l'aide d'un volant.

I - 3 Identification du processus pilote

L'identification a pour but l'obtention d'un modèle mathématique susceptible d'être exploité en terme de commande, c'est-à-dire que le modèle doit être précis sur une période donnée (généralement assez courte) après une initialisation correcte, et ceci malgré sa structure volontairement très élémentaire.

Deux approches sont possibles : on peut, à partir des propriétés physiques du processus, déterminer un certain nombre d'équations caractéristiques du système (4), de structure différentielle. On peut également élaborer un modèle mathématique à partir de la connaissance du comportement du processus vis-à-vis d'excitations appliquées à ses entrées. Les deux types de méthodes sont utilisées ici complémentirement.

I - 3 - 1 Méthode

La méthode d'identification employée se décompose de la manière suivante :

A - On pose un certain nombre d'équations, qui nécessitent des hypothèses intuitives (pouvant être fournies par une connaissance à priori du système).

B - On fait des essais sur le processus, permettant de valider ou de modifier la structure initiale. Ces essais peuvent être indiciels (de part la simplicité de la mise en oeuvre), et permettent d'obtenir certains coefficients.

C - On effectue une synthèse en simplifiant les résultats des deux points précédents. Ceci peut être fait en traçant les fonctions de transfert trouvées en (B) dans le plan de BODE et en cherchant une approximation en module et en phase.

D - Le résultat est alors testé et amélioré à l'aide d'un système analogique fonctionnant simultanément en parallèle avec le processus.

I - 3 2 Etude détaillée

A - Etude théorique des phénomènes thermiques (7)

Une étude théorique doit tenir compte :

- Des phénomènes de combustion
- Des phénomènes d'échange thermique
- Du principe de conservation de l'énergie
- De la structure géométrique du processus.

Considérant le schéma de la figure 9, on voit que le flux calorifique issu de la combustion (q_c) se répartit suivant le système évaporatoire (ϕ_1) et le surchauffeur (ϕ_2). Nous savons, par l'étude du cycle eau-vapeur du pilote (§ I - 1 - 3 Fig. 5), que la surchauffe de la vapeur s'effectue suivant une transformation isobare. Nous pouvons conclure : premièrement que ϕ_1 établit une certaine pression et une température correspondante (température de saturation), deuxièmement que ϕ_2 intervient uniquement dans le réglage de la température, réglage que l'on peut affiner à l'aide de la désurchauffe.

L'équation énergétique du système évaporatoire s'écrit :

$$\dot{E}_V = h_E q_E - h_B q_V + \phi_1 \quad \text{avec } \phi_1 = k_1 q_c$$

D'où l'on peut déduire :

$$\dot{P} = a q_c - b q_V = a q_c - b q_s + b q_d$$

En supposant que, d'une part, l'égalité $q_E = q_V$ est réalisée à Enthalpie constante par la régulation de niveau, d'autre part que $E_V = k p$

Pour obtenir la pression de vapeur surchauffée, il faut introduire

les pertes de charge dans le surchauffeur :

$$p_s = p_B - p_c$$

p est pratiquement proportionnel au carré du débit de vapeur, (loi de Bernouilli), soit :

$$p_s = p_B - c q_s^2$$

De même, l'équation thermodynamique du surchauffeur-désurchauffeur s'écrit :

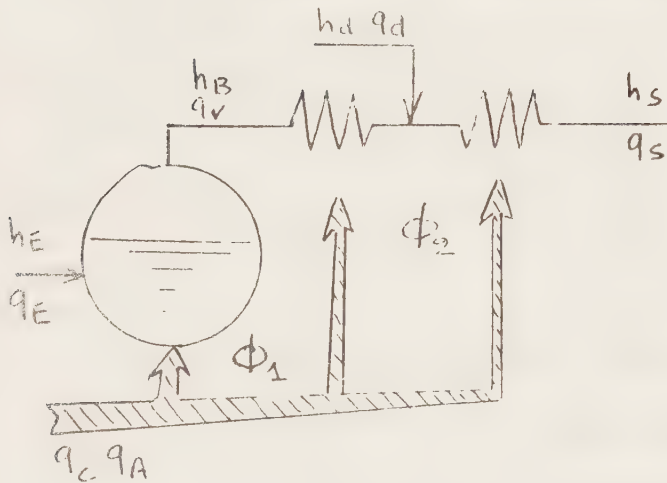
$$\dot{E}_s = h_b q_v + h_d q_d - h_s q_s + \phi_2$$

avec :

$$\phi_2 = k_2 q_c$$

Soit :

$$\dot{E}_s = f(p_B, p_s, t_s, q_c, q_d, q_s)$$



h_B enthalpie de la vapeur saturée
 h_d " de l'eau développée
 h_s " de la vapeur surchauffée
 h_E " de l'eau alimentaire

fig. 9

On cherche à obtenir une expression aussi simple que possible pour le second membre de cette équation. De plus on peut émettre comme hypothèse que la variation d'énergie interne dans le surchauffeur, est proportionnelle à la variation de la température.

Donc : $\dot{t}_s = k \dot{E}_s$

Le problème précédent réside maintenant dans la validation ou l'éventuelle modification de f .

B - Identification expérimentale

Cette identification se fait essentiellement par essais indiciels, dont la mise en oeuvre a été étudiée d'une manière approfondie par E. D. F. (5) (6).

La méthode mise au point par E.D.F. permet de déterminer une fonction de transfert, qui est donnée sous la forme :

$$\frac{1}{(1 + T_p)^n}$$

n étant fractionnaire.

Les perturbations (engénéral des échelons) étant appliquées aux différentes entrées (q_c , q_d , q_s), les réponses de sorties (p et t) permettent d'établir un ensemble de fonctions de transfert, appelé matricé de transfert, donnant sous forme analytique une configuration globale du processus.

C - Synthèse

Cependant, cette méthode d'analyse, de fait de phénomènes thermiques complexes (voir problème de la régulation du niveau d'eau dans le ballon, (§I-2-1, alinéa C), conduit à des ordres relativement élevés (parfois > 4).

En traçant chaque fonction de transfert dans le plan de BODE, on cherche une approximation (en module et en phase) d'un ordre moins élevé. Cette approximation peut donner un déphasage important, compensé, s'il le faut, par l'introduction d'un retard pur.

Cette méthode a permis de constater que la structure choisie en A est insuffisante pour le terme en q_d .

D - Elaboration d'un modèle analogique :

La stratégie est la suivante :

On applique sur les entrées du système, en boucle ouverte, un certain nombre de valeurs de commandes, voisines des commandes optimales. Les sorties p et t sont enregistrées sous forme de valeurs numériques échantillonnées, et sous forme de courbes continues (on enregistre également les entrées).

L'étude de ces courbes permet dans un premier temps de contrôler la structure du modèle mise au point en (C) et de vérifier certains coefficients et les constantes de temps.

Ce modèle mathématique est câblé sur un calculateur analogique.

Un programme hybride (numérique-analogique) permet alors, d'une part d'injecter à l'entrée du processus simulé les valeurs précédemment appliquées au processus réel, d'autre part d'enregistrer sur un traceur de courbes, sorties réelles et sorties simulées.

La comparaison entre ces sorties amène à peaufiner les coefficients du modèle mathématique.

Lorsque les réponses coïncident sensiblement dans un intervalle donné (temps de réponse ou temps d'optimisation par exemple), le modèle peut être considéré comme satisfaisant.

Cette méthode est une version non automatisée de la méthode du modèle.

I - 3 - 3 Résultat

Le modèle mathématique ainsi déterminé et que nous allons utiliser dans la suite est le suivant :

$$\begin{aligned} 3600 \dot{p}_B &= 1,224 \dot{q}_c - 0,10116 \dot{q}_d \\ \dot{p}_s &= p_B - 1,07 \frac{\dot{q}_s}{(1000)^2} \\ 150 \dot{t}_s &= -t_s + 0,8 (\dot{q}_c - 80) - 0,1 (\dot{q}_s - 1000) - z \\ 50 \dot{z} &= -z + 0,8 (\dot{q}_d - 40) \end{aligned}$$

p_s et t_s étant pris par rapport à 20 bars et 300°C.

I - 4 Conclusion

Nous concluons ce chapitre en mettant en évidence deux points fondamentaux : validité du pilote et identification du processus.

Vis à vis du problème essentiel de régulation des centrales, constitué par la commande de deux grandeurs couplées (pression et température de vapeur surchauffée) notre unité pilote possède les mêmes caractéristiques qu'une unité industrielle. Nous avons établi des régulations de type classique pour le niveau et la combustion tandis que nous consacrerons l'optimisation à la chaîne pression - température.

Une méthode d'identification simple et procédant à la fois des principes théoriques et de l'expérience a permis d'établir un modèle de régulation sous la forme de trois équations d'Etat.

LA COMMANDE SOUS - OPTIMALE . THEORIE

Nous développons dans ce chapitre une méthode de calcul d'une commande sous-optimale. Cette méthode utilise des informations fournies par un minimum de solutions exactes du problème de commande optimale, ces solutions étant obtenues hors ligne à partir du modèle mathématique élaboré par identification du processus (cf. chapitre précédent).

Donnons un exemple:

Soit un système d'équations d'état $\dot{x} = u$ avec $u(t) \leq B = \text{Cte}$ (1)
et un critère $J = \frac{1}{2} \int_t^T (x^2 + \alpha^2 u^2) dt + \frac{1}{2} l x^2(T)$ T fixé, $l = \text{Cte}$

A l'aide de quelques solutions exactes (constituant un échantillon suffisamment représentatif des situations plausibles), obtenues à partir du Principe du Maximum par exemple, on désire déterminer une fonction explicite h telle que:

$$u = h (t, x(t), \alpha, B, T) = h (\underline{P})$$

La commande est ici effectuée en boucle fermée et l'utilisateur peut modifier à son gré les paramètres de régulation α et T , ainsi que la contrainte B .

Dans une première partie, nous exposons la caractérisation choisie pour h , en nous référant à une méthode utilisée en reconnaissance de formes et ayant fait l'objet d'applications à la commande optimale pour des critères en temps minimal (2), conduisant à la classification des situations plausibles en deux catégories.

Une deuxième partie est consacrée à l'extension de la méthode au cas où le nombre de catégories est supérieur à deux.

L'élaboration des coefficients de h est obtenue dans une troisième partie à l'aide d'algorithmes d'apprentissage.

Enfin, un quatrième paragraphe est consacré à un exemple d'application des différents algorithmes à un problème simple.

II - 1 Caractérisation de h.

II -1-1 Reconnaissance de forme et systèmes à apprentissage

Considérons le problème suivant : soit à classer des éléments $\underline{x}$ d'un ensemble en deux catégories. Pour cela, déterminer une fonction h telle que :

$h(\underline{x}) > 0$ si $\underline{x}$ appartient à la catégorie 1

$h(\underline{x}) < 0$ si $\underline{x}$ appartient à la catégorie 2

$\underline{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix}$ étant un élément (pour un système de commande, $\underline{x}$ est le vecteur noté $\underline{p}$ ci-dessus, et les catégories correspondent par exemple à $u = \pm 1$).

(une interprétation géométrique de ce problème est que cette classification sépare l'espace à N dimensions de coordonnées $x_1 \dots x_N$ en deux sous-espaces, l'un contenant les éléments $\underline{x}$ de la catégorie 1, l'autre contenant les éléments de la catégorie 2. $h(\underline{x}) = 0$ est l'hypersurface de séparation.)

La résolution de ce problème revient à réaliser un générateur de fonction tel qu'à partir d'un nombre minimum d'éléments $\underline{x}$, sa structure interne lui permette de classer correctement l'ensemble des éléments susceptibles d'être rencontrés, ou du moins le plus grand nombre possible de ces éléments. Autrement dit, le générateur de fonction doit " apprendre " à " reconnaître " l'hypersurface $h(\underline{x}) = 0$.

La simplicité de mise en oeuvre et l'expérience acquise dans les systèmes de commande par tout ou rien (2) nous ont amené à choisir comme élément de base de ce générateur de fonction, l'élément linéaire à seuil.

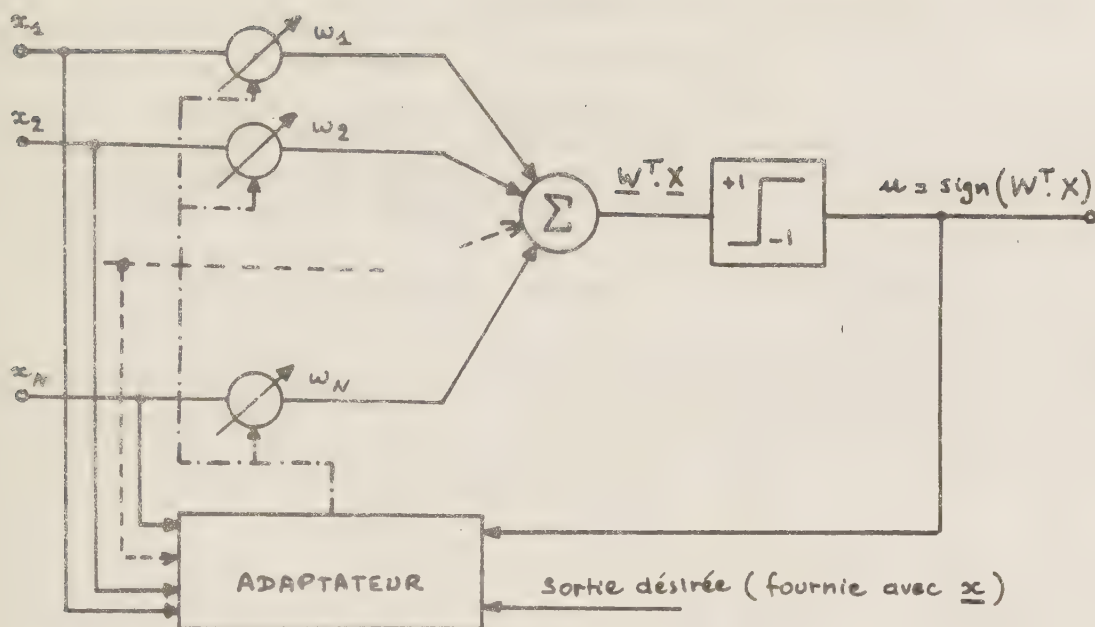
II - 1-2 L'élément linéaire à seuil (3) (4)

A - Description

La figure 9 montre un tel élément, les entrées (analogiques ou binaires) sont les composantes de $\underline{x}$, $x_1 \dots x_N$. La sortie analogique est la somme pondérée des entrées, soit :

$$\hat{h}(\underline{x}) = \underline{w}^T \cdot \underline{x} \quad (\text{II-a})$$

$\underline{w} = \begin{pmatrix} w_1 \\ \vdots \\ w_N \end{pmatrix}$ est un vecteur de pondération, inconnu, dont les composantes (poids), analogiques, sont les coefficients à adapter de l'élément linéaire.



- Figure 9 -
bis

la sortie binaire de l'élément est $+1$ si $h(\underline{x})$ est positif
($\underline{x}$ appartient à la catégorie 1), -1 si $h(\underline{x})$ est < 0
($\underline{x}$ appartient à la catégorie 2)

Pour un $\underline{x}$ donné, l'adaptateur (cf. Fig. 9) compare la sortie réelle et la sortie désirée (c'est-à-dire ici la catégorie $+1$ et -1) fournies avec chaque $\underline{x}$. Les poids sont modifiés en fonction de cette comparaison et ceci pour chaque élément $\underline{x}$. Cette adaptation des poids est effectuée à l'aide d'algorithmes d'apprentissage que nous mentionnons plus loin.

L'apprentissage continu jusqu'à ce que tous les échantillons $\underline{x}$ soient correctement classés (ou que l'erreur de classement ait atteint une valeur minimum prédéterminée).

B - Choix du type d'entrées

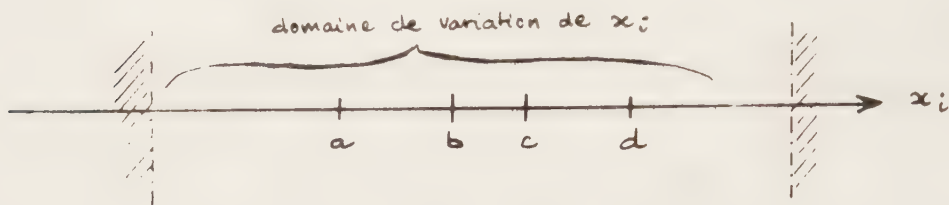
Nous avons dit que les entrées de l'élément linéaire pouvaient être analogique ou binaires. En fait, il est avantageux que les entrées soient binaires, pour une raison de simplicité d'algorithme de calcul du vecteur poids $\underline{w}$.

De plus, du point de vue du problème de commande, il n'y a pas d'inconvénient à quantifier les variables d'entrées du fait de l'incertitude existant sur les mesures de (y) et de la faible sensibilité des trajectoires optimales aux variations des paramètres de régulation (α et T).

C - Codages des variables d'entrée

Soit par exemple une grandeur x_i (figure 10) . Le codage de cette grandeur consiste à quantifier le domaine plausible de variation de x_i en un certain nombre de zones et à représenter chacune de ces zones par un nombre binaire. La largeur de ce nombre binaire sera au plus égale au nombre de zones définies par la quantification de x_i .

Ce nombre binaire est appelé " vecteur partiel " .



- figure 10 -

Ce code pourra avoir la structure suivante : (5)

$x_i \leq a$	(1 0 0 0 0)
$a < x_i \leq b$	(0 1 0 0 0)
$b < x_i \leq c$	(0 0 1 0 0)
$c < x_i \leq d$	(0 0 0 1 0)
$d < x_i$	(0 0 0 0 1)

Dans l'exemple considéré, chaque vecteur partiel (noté $\underline{x}_i$) constitue un état à présenter à un élément linéaire à cinq entrées.

Si nous reprenons le problème posé au paragraphe II.1.1, chaque composant x_i du vecteur $\underline{x}$ est représentée par un nombre binaire $\underline{x}_i$ défini suivant un certain code. La longueur de ce nombre ne dépend que de la quantification choisie a priori; par exemple, x_1 sera quantifié en cinq zones, x_2 en six zones, x_3 en 4 zones, etc x_N en 3 zones . Chaque vecteur $\underline{x}$ sera donc représenté par un nombre binaire $\underline{x}$ de la forme :

$$\underline{x} = \left\{ \begin{array}{c|c|c|c|c} 0 & 1 & 0 & 0 & 0 \\ \hline x_1 \end{array} \right\} \left\{ \begin{array}{c|c|c|c|c|c} 0 & 0 & 0 & 1 & 0 & 0 \\ \hline x_2 \end{array} \right\} \text{-----} \left\{ \begin{array}{c|c|c|c} 0 & 0 & 0 & 1 \\ \hline x_i \end{array} \right\} \text{-----} \left\{ \begin{array}{c|c|c} 1 & 0 & 0 \\ \hline x_N \end{array} \right\}$$

II - 1-3 Propriétés de l'Elément linéaire.

Nous décrivons quelques unes de ces propriétés à partir d'un exemple pris dans l'espace à trois dimensions, dont la visualisation est aisée.

La figure 11-a montre le schéma d'un élément linéaire à seuil à trois entrées. Pour simplifier le problème, nous supposons que chaque entrée ne peut prendre que deux valeurs : 0 ou 1. Sur la figure 11-b, sont représentées toutes les combinaisons possibles de ces trois entrées, chaque combinaison étant affectée à un sommet d'un cube (il existe $2^3 = 8$ combinaisons possibles). Si nous reprenons la notion de code, définie au paragraphe précédent, qui est liée à la notion de quantification, nous voyons que chaque sommet représente une zone complète de l'espace à trois dimensions. Nous affectons maintenant à chaque combinaison une indication de catégorie que nous notons +1 ou -1 (figure 11-c). La classification correcte en ces deux catégories exige que l'espace soit séparé en deux sous-espaces, d'une manière analogue à celle indiquée sur la figure.

A - Une première remarque s'impose, qui définit une propriété de l'élément linéaire : Dans l'hypothèse où on choisit un plan Π comme critère de classement, il existe une infinité de solutions correspondant à une infinité de positions de ce plan, qui donnent toutes un classement correct.

B - L'équation de chacun de ces plans donne les poids de l'élément linéaire. Cette équation est de la forme :

$$\underline{x}^T \cdot \underline{w} = w_1 x_1 + w_2 x_2 + w_3 x_3 + k = 0 \quad (= h(\underline{x}))$$

Pour les combinaisons appartenant à la catégorie 1 (+1), le signe de cette fonction est positif. Pour les combinaisons appartenant à la catégorie 2 (-1), le signe est négatif.

C - Pour une classification donnée, on peut trouver deux plans parallèles qui réalisent ce classement et qui ont une distance maximum. La distance d entre ces deux plans intervient de deux manières :

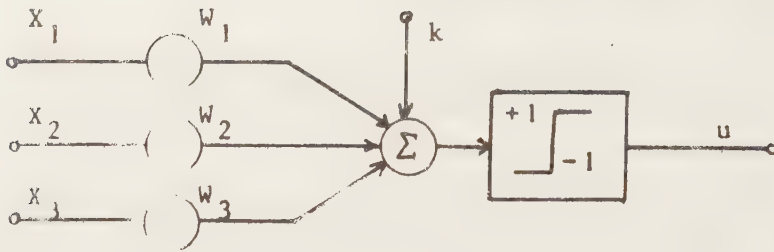
1- Cette distance d intervient dans l'apprentissage. Plus elle est faible et plus le classement est difficile à réaliser.

2- Elle intervient dans la sensibilité de l'élément linéaire aux signaux bruités, suivant que le niveau de ce bruit est de l'ordre, ou non, de la distance d .

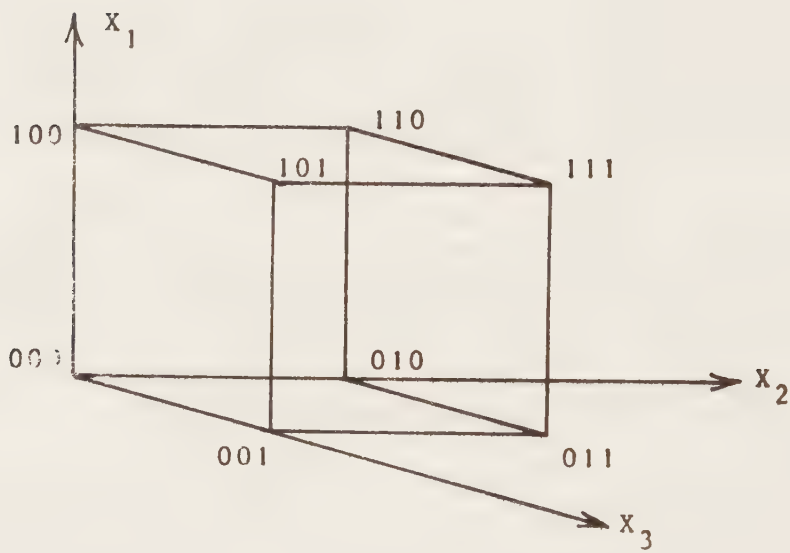
D - L'apprentissage se déroule de la manière suivante :

Initialement, les valeurs de w_1 , w_2 , w_3 sont quelconques, ce qui signifie que le plan Π occupe dans l'espace une position quelconque. La modification des poids intervient dans la mesure où chaque échantillon présenté est incorrectement classé par le plan.

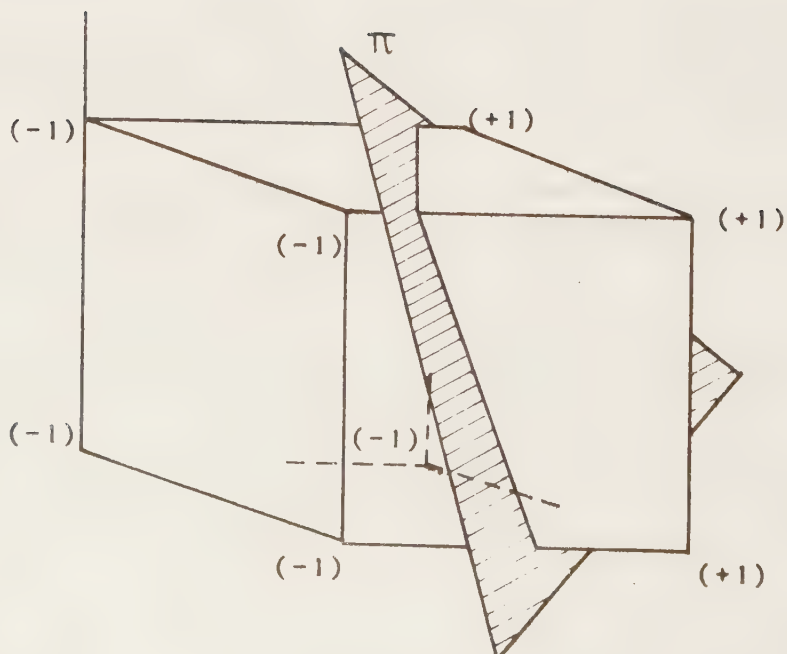
Cependant, on ne peut pas réaliser, à l'aide du seul plan Π , tous les classements possibles. Supposons par exemple que les combinaisons (000) et (111) appartiennent à la même catégorie 1, alors que les six autres combinaisons appartiennent à la catégorie 2. On voit aisément qu'il n'est pas possible de séparer ces deux catégories à l'aide d'un seul plan. Les classements que l'on peut réaliser à l'aide d'un seul plan sont appelés linéairement séparables.



- 11 a -



- 11 b -



- FIGURE 11 -

- 11 c -

II - 4 Choix d'un code

Parmi l'ensemble des vecteurs $\underline{x}$ possible, soient Q et R le nombre d'éléments plausibles, codés, appartenant respectivement à la catégorie 1 ($u = +1$) et à la catégorie 2 ($u = -1$) et soient q et r tels que :

$$q < Q$$

$$r < R$$

Les q éléments de la catégorie 1 et les r éléments de la catégorie 2 représentent des échantillons de l'ensemble de ces éléments Q et R

Soit P^+ la matrice qui a pour lignes les q échantillons codés de la catégorie 1, P^- la matrice qui a pour lignes les r échantillons de la catégorie 2.

Le classement de ces échantillons sera correct si :

$$\begin{aligned} P^+ \cdot \underline{w} &> 0 \\ P^- \cdot \underline{w} &< 0 \end{aligned} \quad (\text{II-b})$$

Définissons maintenant la matrice A telle que :

$$A = \begin{bmatrix} P^+ \\ -P^- \end{bmatrix}$$

Cette écriture permet de remplacer les deux équations (II-b) par une nouvelle équation, plus générale :

$$A \cdot \underline{w} > 0 \quad (\text{II-c})$$

L'équation (II-c) est l'"Equation d'Apprentissage".

Théorème : Le générateur de fonction adaptatif ne pourra réaliser une solution de l'équation (II-c) que si les éléments (lignes) codés de la matrice A sont linéairement indépendants.

Démonstration : Reprenons le schéma du générateur de fonction tel qu'il a été défini Fig. 9. Chaque composante x_i du vecteur $\underline{x}$ est quantifiée en un ensemble de vecteurs partiels $\underline{X}_i$. Chaque sortie partielle est la somme pondérée de ces entrées, soit :

$$R_i(\underline{x}_i) = \underline{x}_i^T \cdot \underline{w}_i \quad (\text{II-d})$$

$\underline{X}_i^T$ étant le vecteur ligne correspondant à x_i quantifié. Ce vecteur ligne est composé de 1 et de 0, suivant le code défini précédemment

$$R_i \text{ étant une fonction partielle telle que } \hat{h} = \sum_{i=1}^N R_i(\underline{x}_i)$$

Soit P la matrice dont les lignes sont tous les vecteurs partiels possibles représentant la composante x_i (leur nombre est égal au nombre de zones de quantification de x_i dans son domaine de variation plausible).

Soit H_i le vecteur ~~colonne~~ dont les composantes sont les valeurs de la fonction partielle $h_i(\underline{x}_i)$ (à un vecteur partiel $\underline{x}_i$ correspond une valeur de h_i).

Nous pouvons écrire l'équation (II-d) sous la forme :

$$P \cdot \underline{w}_i = H_i(\underline{x}_i)$$

dont la solution est : $\underline{w}_i = P^{-1} \cdot H_i(\underline{x}_i) \quad \forall H_i(\underline{x}_i)$

or, pour que P^{-1} existe, il faut et suffit que chaque ligne de P (vecteurs partiels) soit linéairement indépendante des autres. Ceci a pour conséquence que le code utilisé soit lui-même linéairement indépendant.

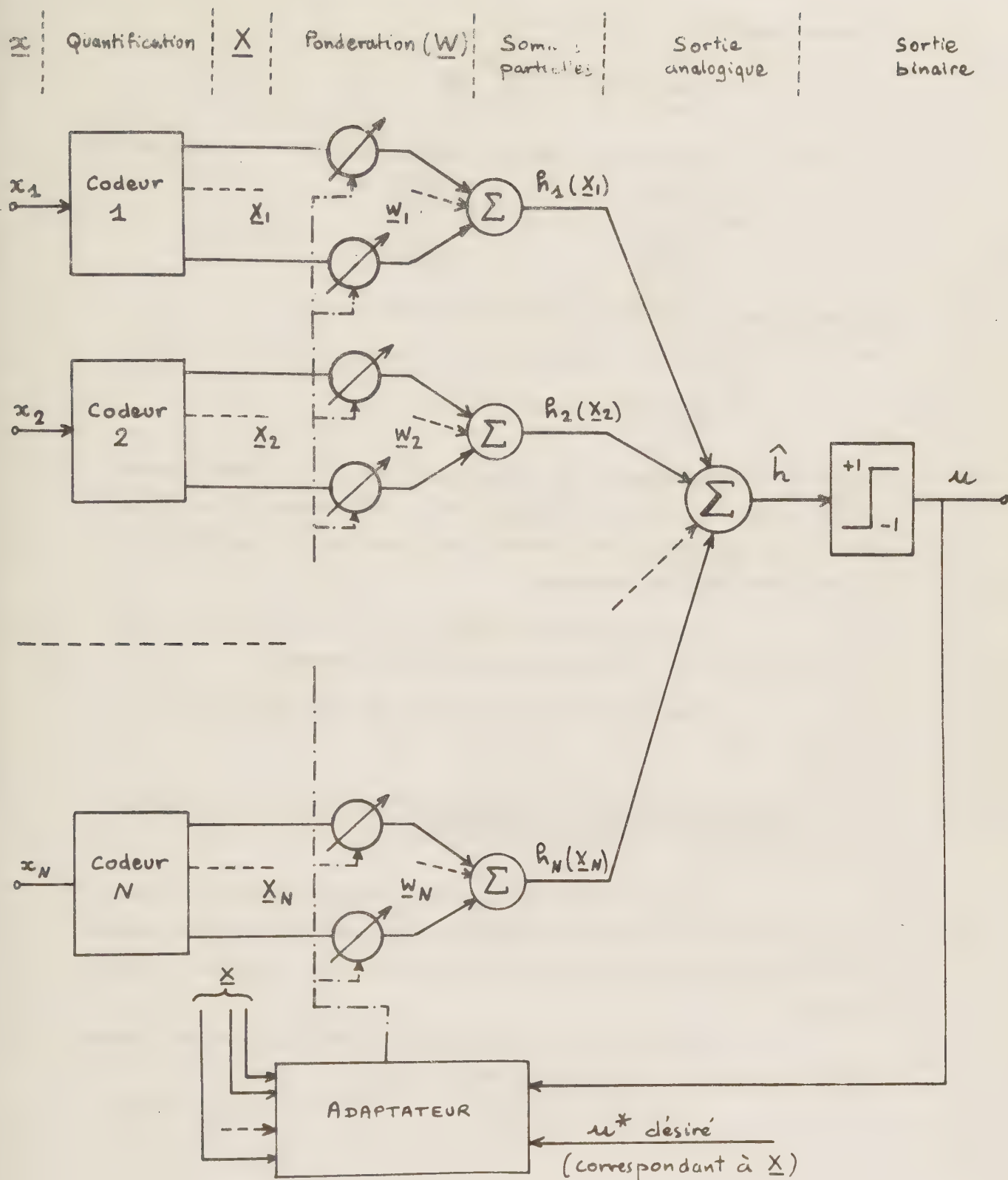
Reprenant l'exemple du paragraphe II-4 Fig. 2, le code suivant n'est pas linéairement indépendant (du fait de la présence du vecteur partiel 0000)

$x_i \leq a$	(1 1 1 1)	(1 1 1 1 1)
$a < x_i \leq b$	(1 1 1 0)	(1 1 1 1 0)
$b < x_i \leq c$	(1 1 0 0)	(1 1 1 0 0)
$c < x_i \leq d$	(1 0 0 0)	(1 1 0 0 0)
$d < x_i$	(0 0 0 0)	(1 0 0 0 0)

le code peut être rendu linéairement indépendant par adjonction d'un 1 à gauche.

II-1-5 Nouvelle caractérisation de l'Elément linéaire généralisé

Ayant choisi de coder les vecteurs $\underline{x}$, nous avons vu que chaque vecteur partiel constituait l'entrée d'un élément linéaire défini au paragraphe II-2. Le schéma complété de l'élément linéaire admettant pour entrée le vecteur $\underline{x}$ est donné Fig. 12 .



ELEMENT LINEAIRE GENERALISE - FIGURE 12 -

La somme pondérée des entrées est :

$$\hat{h} = \sum_{i=1}^N R_i(\underline{x}_i) = \underline{x}^T \cdot \underline{w}$$

avec : R_i fonction, linéaire ou non, de $\underline{x}_i$ ("fonction partielle")

$$\underline{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_N \end{bmatrix} \quad \underline{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix}$$

La sortie binaire est une indication de catégorie :

$$\hat{h}(\underline{x}) > 0 \quad \text{catégorie 1 (u = + 1)}$$

$$\hat{h}(\underline{x}) < 0 \quad \text{catégorie 2 (u = - 1)}$$

$\hat{h}(\underline{x})=0$ est l'approximation de l'hypersurface $h(\underline{x})=0$; Cette approximation est due au fait que l'espace des $\underline{x}$ a été quantifié en hypercubes, dont les dimensions donnent d'ailleurs une indication sur l'erreur systématique commise.

Le calculateur, composé d'éléments linéaires, sera appelé dans la suite "générateur de fonction adaptatif"

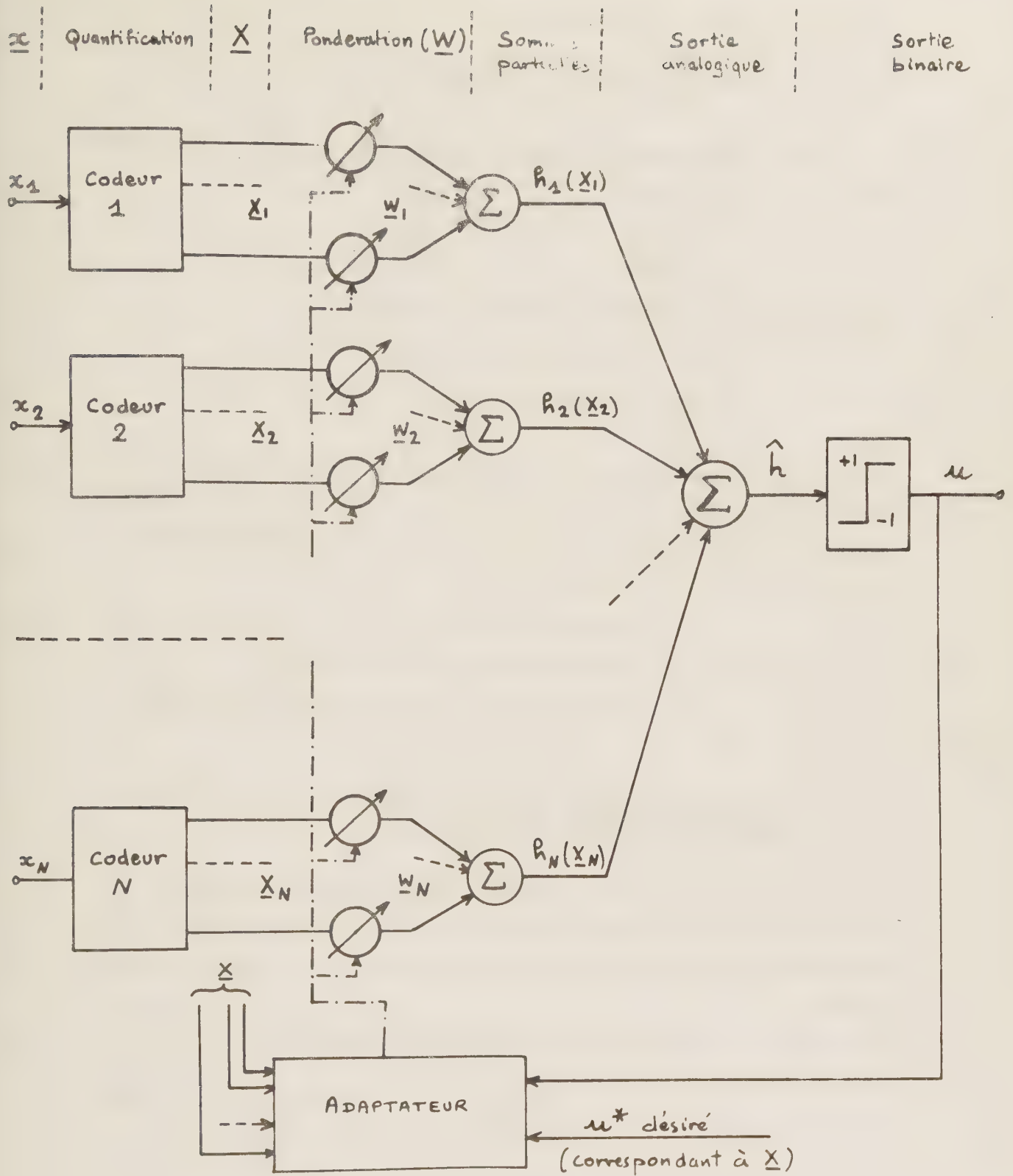
Les vecteurs $\underline{x}$ seront appelés des "Echantillons".

II - 2 CAS OU LE NOMBRE DE CATEGORIES EST > 2 (COMMANDE CONTINUE QUANTIFIEE FINEMENT PAR EXEMPLE)

Nous avons jusqu'à présent étudié un générateur de fonction qui peut classer des éléments d'un ensemble en deux catégories. Nous allons montrer sur un exemple qu'il est possible de généraliser.

Considérons un problème de classification en 3 catégories (en termes de commande 3 valeurs de u)

Soit un plan défini par ses axes X_1 et X_2 (Fig.13)



ELEMENT LINEAIRE GENERALISE - FIGURE 12 -

La somme pondérée des entrées est :

$$\hat{h} = \sum_{i=1}^N h_i(\underline{x}_i) \cdot \underline{x}^T \cdot \underline{w}$$

avec : h_i fonction, linéaire ou non, de $\underline{x}_i$ ("fonction partielle")

$$\underline{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_N \end{bmatrix} \quad \underline{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix}$$

La sortie binaire est une indication de catégorie :

$$\hat{h}(\underline{x}) > 0 \quad \text{catégorie 1 (u = + 1)}$$

$$\hat{h}(\underline{x}) < 0 \quad \text{catégorie 2 (u = - 1)}$$

$\hat{h}(\underline{x})=0$ est l'approximation de l'hypersurface $h(\underline{x})=0$; Cette approximation est due au fait que l'espace des $\underline{x}$ a été quantifié en hypercubes, dont les dimensions donnent d'ailleurs une indication sur l'erreur systématique commise.

Le calculateur, composé d'éléments linéaires, sera appelé dans la suite "générateur de fonction adaptatif"

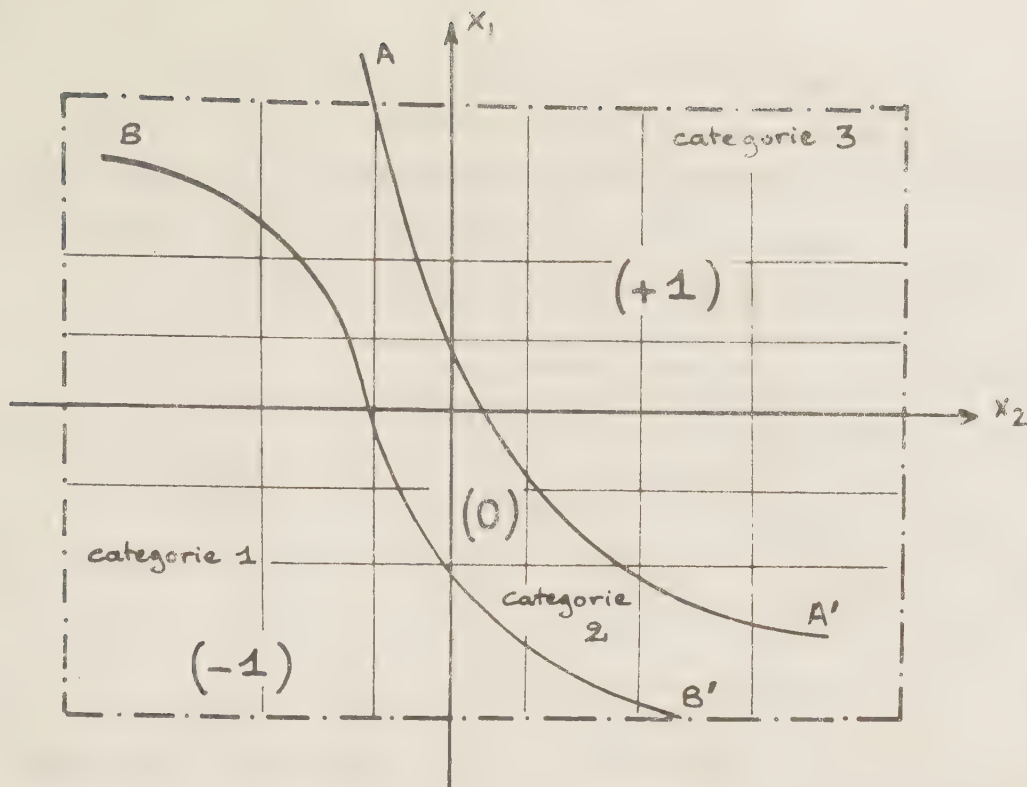
Les vecteurs $\underline{x}$ seront appelés des "Echantillons".

II - 2 CAS OU LE NOMBRE DE CATEGORIES EST > 2 (COMMANDE CONTINUE QUANTIFIEE FINEMENT PAR EXEMPLE)

Nous avons jusqu'à présent étudié un générateur de fonction qui peut classer des éléments d'un ensemble en deux catégories. Nous allons montrer sur un exemple qu'il est possible de généraliser.

Considérons un problème de classification en 3 catégories (en termes de commande 3 valeurs de u)

Soit un plan défini par ses axes X_1 et X_2 (Fig.13)



- figure 13 -

Les deux courbes AA' et BB' délimitent les catégories.

Caractérisons la catégorie 1 par $u = -1$

la catégorie 2 par $u = 0$

la catégorie 3 par $u = +1$

Les valeurs plausibles de x_1 et x_2 sont encadrées par un trait mixte. A l'intérieur de ce domaine, chaque point peut être représenté par un vecteur $\underline{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ et l'indication de sa catégorie, mais aussi par le vecteur $\underline{x}$ et sa position par rapport aux deux courbes AA' et BB' .

Chacune de ces courbes partage le plan en deux sous-catégories :

A gauche de AA' , la catégorie peut être définie par -1 ,
à droite par $+1$

A gauche de BB' , la catégorie peut être définie par -1 ,
à droite par $+1$

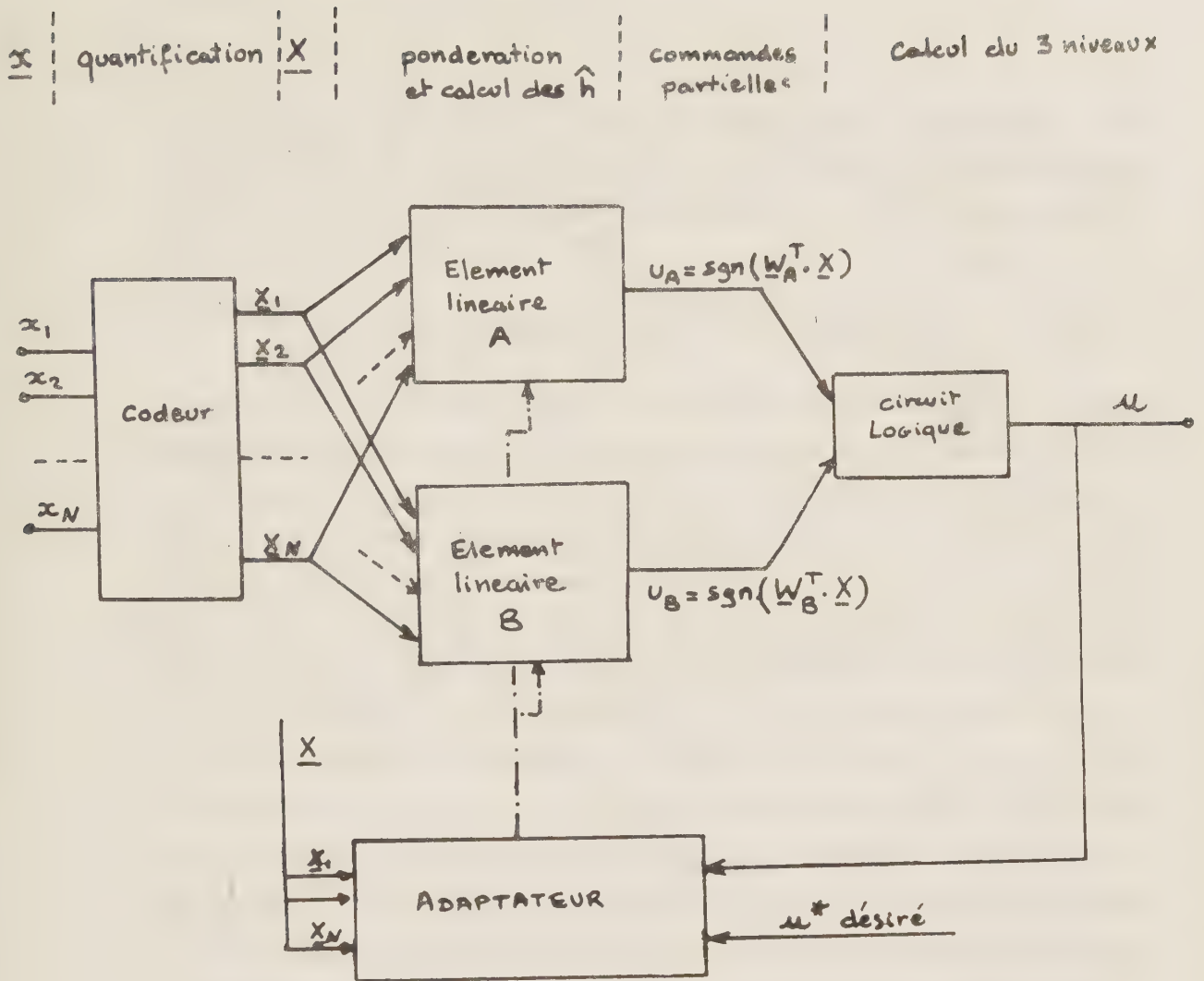
Ceci donne l'idée d'employer deux générateurs de fonction adaptatifs, le premier "reconnaissant" AA', le deuxième reconnaissant BB', les sorties binaires étant respectivement u_A et u_B .

La caractérisation réelle, u , peut être retrouvée à partir de u_A et u_B d'après la table de vérité suivante :

:	:	:	:	:
:	u_A	u_B	u	catégorie
:	- 1	- 1	- 1	1
:	- 1	+ 1	0	2
:	+ 1	- 1	0	2
:	+ 1	+ 1	+ 1	3
:	:	:	:	:

Cet exemple montre que l'on peut classer des éléments en 3 catégories à l'aide de deux générateurs de fonction suivant le schéma de la Fig. 14

On peut appliquer la méthode précédente pour une classification en n catégories.



- FIGURE 14 -

II - 3 MISE EN OEUVRE DU GENERATEUR DE FONCTION ADAPTATIF

II-3-1 Choix et présentation des échantillons

Dans un problème réel, le nombre d'échantillons possibles est considérable (au minimum, ce nombre d'échantillons possibles est égal au nombre d'hypercubes. Par exemple, dans un problème où $\underline{x}$ a 5 composantes, dont la première est quantifiée en 5 zones, la deuxième en 6 zones, la troisième en 5 zones, la quatrième en 4 zones, la cinquième en 5 zones, nous aurons $5 \times 6 \times 5 \times 4 \times 5 = 3\ 000$ hypercubes, et donc au minimum 3 000 échantillons possibles).

On sera donc amené à ne considérer qu'un nombre restreint d'éléments $\underline{x}$ (par exemple quelques centaines). On peut choisir les hypercubes qui bordent de part et d'autre l'hypersurface $h = 0$, mais ceci n'est valable que pour un espace de dimensions $N \leq 3$, dans lequel on peut avoir une idée de la forme de cette surface, sinon de son équation analytique (du moins pour les problèmes en temps minimal). Lorsque $N > 3$, il faut choisir les échantillons dont une connaissance à priori du système a montré la fréquence élevée d'apparition (et pour lesquels on connaît également la catégorie, autrement dit la commande optimale exacte associée).

Mais le fait de classer correctement les échantillons choisis ne signifie pas que l'ensemble des éléments sera correctement classé. Ceci débouche sur la notion de généralisation, qui est la possibilité pour un générateur de fonctions de classer correctement des éléments qui n'ont pas été utilisés pour son apprentissage. Un certain nombre de règles (4) (6) permettent de prédire, du moins théoriquement, si un générateur de fonctions pourra généraliser ; la plus commune étant la condition pour $h = 0$ d'être monotone suivant un au moins de ces paramètres $\left(\frac{\partial h}{\partial x_i} \geq 0 \text{ ou } \leq 0 \right)$ et dans le domaine de variation plausible de ce paramètre.

Même si ces règles ne sont pas vérifiées (ou ne sont pas applicables, étant donné l'importance du processus), le générateur de fonctions tel qu'il a été défini précédemment à tendance à généraliser, comme nous l'avons vu lors d'exemples d'application.

De part la nature du générateur adaptatif et pour simplifier l'algorithme d'apprentissage (c'est-à-dire l'encombrement de l'ordinateur), on présente à son entrée un seul vecteur $\underline{x}$ à la fois et l'ordre de présentation est aléatoire. La modification du vecteur de pondération $\underline{W}$ n'intervient que dans la mesure où l'échantillon est incorrectement classé. Nous allons étudier la méthode de classement dans le paragraphe suivant.

II - 3,2 - Méthodes d'apprentissage - Les algorithmes

Les méthodes utilisées emploient toutes un algorithme d'apprentissage et sont itératives. Elles sont appliquées jusqu'à l'obtention d'une solution (le critère d'arrêt étant le classement correct de tous les échantillons, ou bien l'atteinte d'un facteur de qualité, fixé à priori).

Les algorithmes d'apprentissage permettent d'ajuster le vecteur $\underline{W}$ de telle sorte que l'équation

$$A \cdot \underline{W} > 0 \quad \text{soit vérifiée}$$

Une remarque : Il existe une infinité de solutions $\underline{W}$. Le choix d'une solution particulière est liée au critère d'arrêt de la méthode d'adaptation. Ce choix peut être dicté par des considérations de rapidité de calcul, ou bien par la nature des composants du générateur de fonction (7) (8).

Si nous reprenons l'image géométrique, les algorithmes permettent de rendre minimum la distance entre l'hypersurface "vraie" $h = 0$ et l'hypersurface quantifiée $\hat{h} = 0$ (cette distance minimum, nous l'avons dit, dépend de la quantification utilisée)

Nous donnons ci-dessous une liste des principaux algorithmes d'apprentissage, avec leur principe d'utilisation et leurs caractéristiques.

- soit $\underline{W}(0)$ le vecteur de pondération initial, non adapté (quelconque, mais $\neq 0$)
- $\underline{W}(k)$ est le vecteur obtenu après k corrections
- a_j concerne le $j^{\text{ème}}$ échantillon présenté

$$\begin{aligned} \underline{a}_j &= \underline{X}_j^T & \text{si } u &= +1 \\ \underline{a}_j &= -\underline{X}_j^T & \text{si } u &= -1 \end{aligned}$$

1 - Algorithme à incrément fixe

$$\begin{aligned} \underline{w}(k+1) &= \underline{w}(k) & \text{si } \underline{a}_j \cdot \underline{w}(k) &\geq 0 \\ \underline{w}(k+1) &= \underline{w}(k) + \frac{\beta}{N} \underline{a}_j^T & \text{si } \underline{a}_j \cdot \underline{w}(k) < 0 \end{aligned}$$

β : quelconque > 0

N : dimension de l'espace de référence (nombre de composantes de $\underline{X}$)

Cet algorithme garantit la convergence vers une solution quel que soit β . (2)

Il est très simple dans sa mise en oeuvre

2 - Algorithme par relaxation

$$\begin{aligned} \underline{w}(k+1) &= \underline{w}(k) & \text{si } \underline{a}_j \cdot \underline{w}(k) &\geq 0 \\ \underline{w}(k+1) &= \underline{w}(k) + \frac{\pi}{N} (1 - \underline{a}_j \cdot \underline{w}(k)) \underline{a}_j^T & \underline{a}_j \cdot \underline{w}(k) < 0 \end{aligned}$$

Cet algorithme assure la convergence vers une solution pour $0 < \pi \leq 2$. L'amélioration par rapport au précédent algorithme de la rapidité de convergence (qui dépend du coefficient $\frac{\pi}{N}$ équilibre l'inconvénient d'une mise en oeuvre délicate (choix de π)).

3 - Algorithme à Bêta variable

Il constitue une amélioration sensible du premier algorithme, car il évolue avec l'état de la convergence :

Soit P_{cc} un facteur de qualité défini comme le rapport du nombre d'échantillons correctement classés après k adaptation au nombre d'échantillons à classer :

$$P_{cc} \leq 1$$

On peut fixer à priori une valeur caractéristique de P_{cc} , par exemple 0.9 (cela signifie que les 9/10 des échantillons sont correctement classés). Cette valeur sera désignée par Δ

$$\begin{aligned} \underline{w}(k+1) &= \underline{w}(k) & \text{si} \quad \underline{a}_j \cdot \underline{w}(k) &\geq 0 \\ \underline{w}(k+1) &= \underline{w}(k) + \beta \underline{a}_j^T & \text{si} \quad \underline{a}_j \cdot \underline{w}(k) &< 0 \end{aligned}$$

$$\beta = \alpha e \quad \text{si} \quad P_{cc} \leq \Delta$$

$$\beta = \frac{|\underline{w}(k)|}{2 P_{cc}^2 N} \quad \text{si} \quad P_{cc} > \Delta$$

Cet algorithme présente tous les avantages de simplicité et de convergence de l'incrément fixe. De plus, il améliore ces qualités en fin de convergence en l'accéléérant.

Remarque : On démontre,

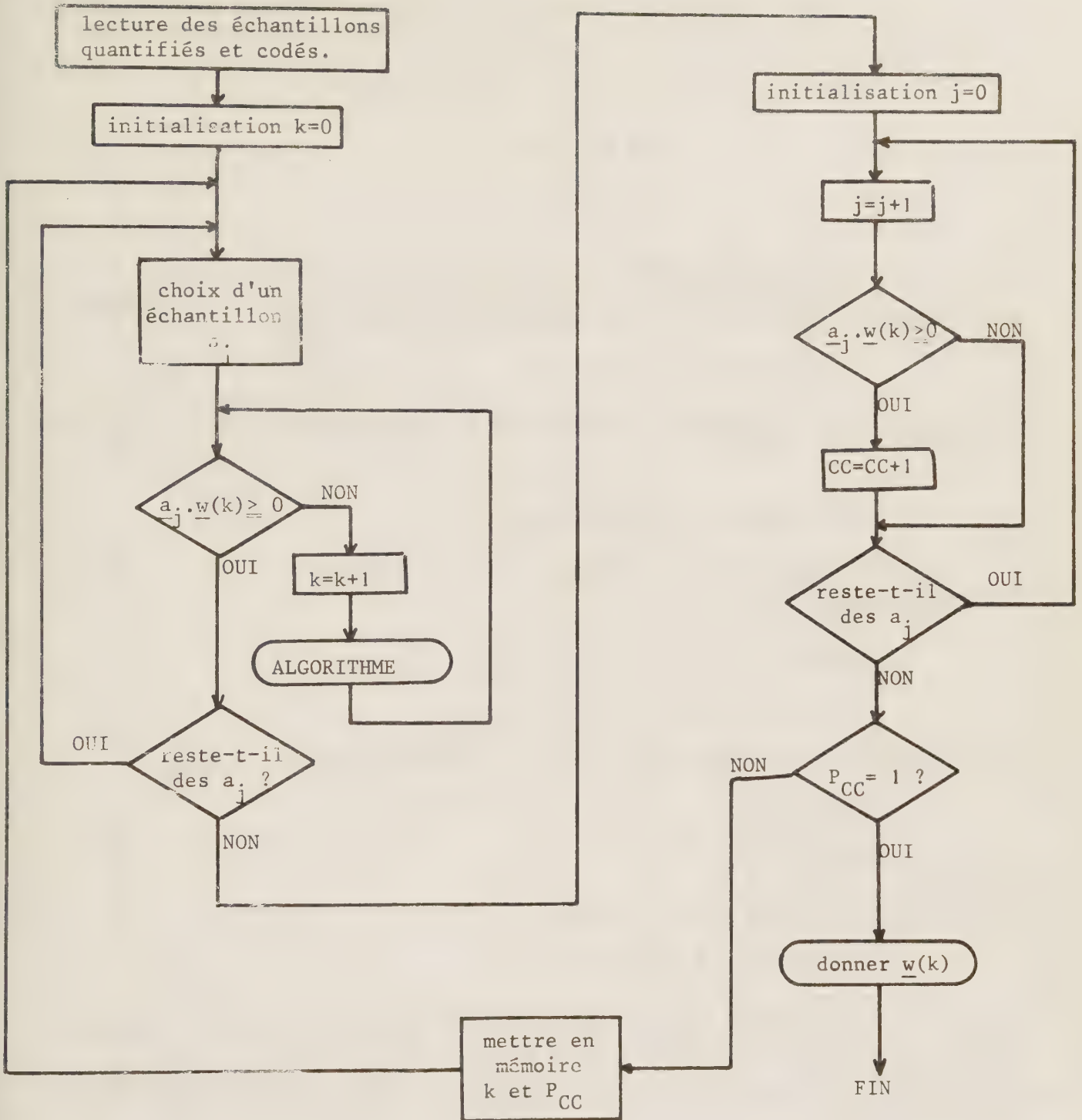
- que l'emploi de ces différents algorithmes est une version discrète de la méthode du gradient (7)
- que ces algorithmes assurent la convergence pour certaines valeurs de leurs paramètres (8)

Pour conclure ce paragraphe, la mise en oeuvre d'un générateur de fonction adaptatif fait appel aux étapes suivantes :

- 1 - quantifier chaque variable (composante du vecteur $\underline{x}$) dans son domaine de variation plausible
- 2 - Caractériser les hypercubes ainsi définis par un code linéairement indépendant
- 3 - Choisir un certain nombre d'échantillons pour lesquels la catégorie est connue (en pratique pour lesquels on connaît la commande optimale exacte correspondante)
- 4 - Choisir un algorithme d'apprentissage (mise en oeuvre simple, convergence rapide)
- 5 - Présenter chaque échantillon à l'entrée du générateur de fonction et la catégorie correspondante à l'entrée de l'adaptateur et appliquer l'algorithme jusqu'à l'obtention d'une solution.

On peut remarquer que cette méthode peut être appliquée sur le générateur de fonction lui-même, ou plus avantageusement être simulée sur ordinateur. L'organigramme de cette simulation est

donné Fig. 15. On peut séparer deux phases : une phase d'apprentissage qui ajuste $\underline{w}(k)$ pour chaque vecteur $\underline{x}$ présenté et une phase de test qui, à partir d'un vecteur $\underline{w}(k)$ calcule P_{cc} défini plus haut. Si $P_{cc} < 1$, on revient à la phase d'apprentissage. Lorsque $P_{cc} = 1$ (ou une valeur quelconque (< 1) prédéterminée) on arrête le processus. Le vecteur poids résultant est considéré comme vecteur à appliquer en ligne.



Simulation de la méthode d'apprentissage
ORGANIGRAMME

II - 4 EXEMPLE D'APPLICATION

Dans cet exemple nous comparons les résultats obtenus par l'application des différents algorithmes décrits précédemment à un problème où la surface de séparation est donnée explicitement.

Soit l'équation $f(x_1, x_2) = x_1^2 + x_2^2 - 0,64$

$$0 \leq x_1 < 0,9$$

$$0 \leq x_2 < 0,9$$

Cette équation définit un secteur circulaire dans le premier quadrant, à l'intérieur duquel on peut affecter une catégorie, correspondant au signe de la fonction f : cette catégorie est définie par

$$u = \text{sgn} \{ f(x_1, x_2) \} < 0 \quad (\text{figure 16})$$

(Bien entendu, la catégorie correspondant à l'extérieur du secteur circulaire est définie par $u > 0$)

F.W. SMITH (4) propose d'écrire :

$$f(x_1, x_2) = f_1(x_1) + f_2(x_2)$$

soit par exemple, ici :

$$f_1(x_1) = x_1^2$$

$$f_2(x_2) = x_2^2 - 0,64$$

puis de chercher une approximation de ces deux fonctions, de la forme :

$$\hat{f}_1(\underline{x}_1) = \underline{x}_1^T W_1$$

$$\hat{f}_2(\underline{x}_2) = \underline{x}_2^T W_2$$

la sortie de l'élément linéaire étant :

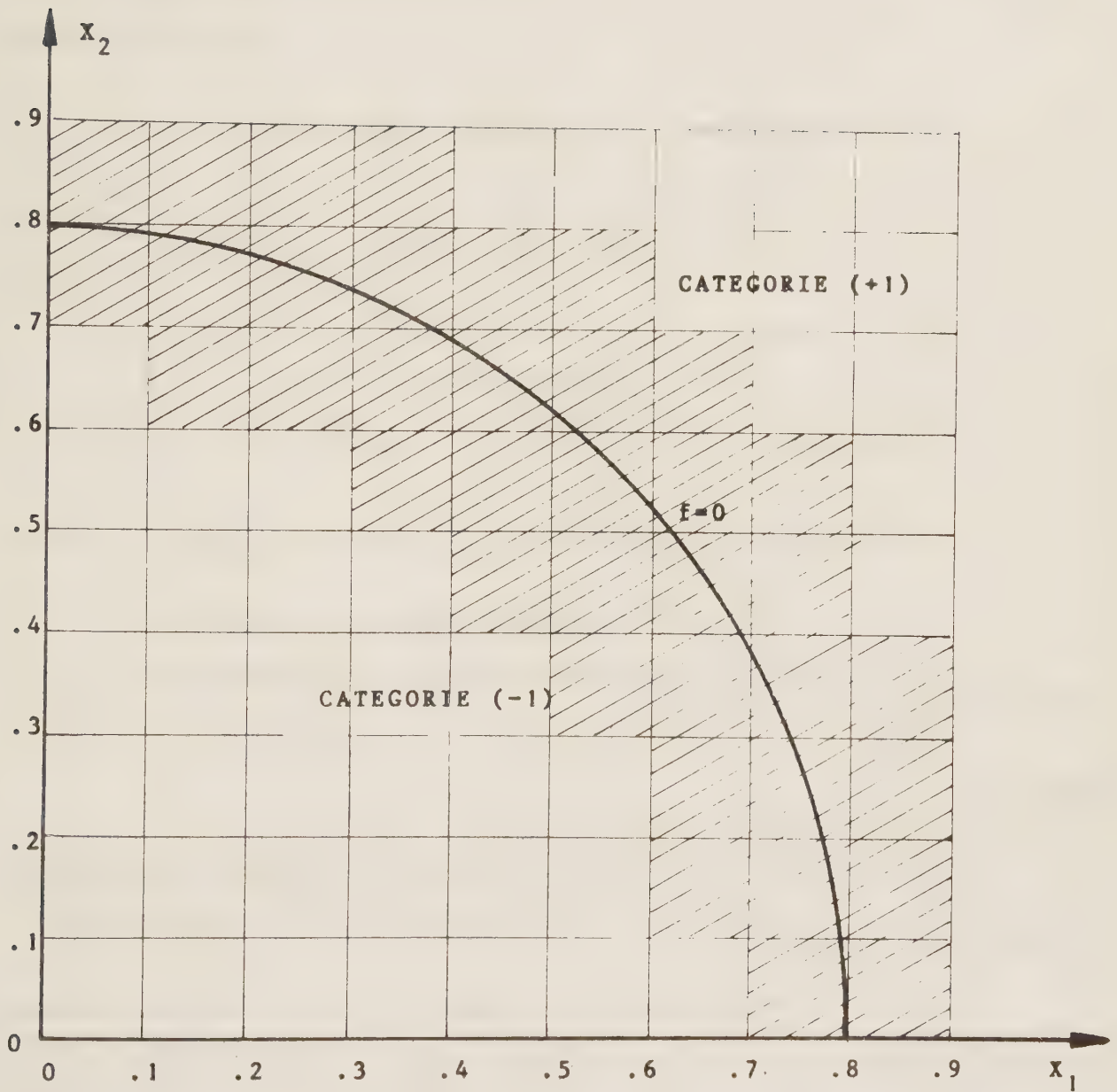
$$\hat{u} = \text{sgn}(\underline{x}_1^T W_1 + \underline{x}_2^T W_2)$$

Cette méthode suppose que les variables sont séparables et en particulier qu'il n'existe pas de produit en croix de la forme $x_i x_j$ dans l'expression de f (ce qui d'ailleurs est le cas dans l'exemple considéré). De plus, la courbe $f = 0$ doit être monotone suivant au moins une des variables x_i ou x_j .

Quantification

L'espace défini par les inégalités plus haut est quantifié en 81 zones (9 en x_1 , 9 en x_2), identiques. Le code choisi pour identifier ces zones est un code mono-élément tel que celui proposé au paragraphe II-1-2-C.

Nous choisissons une valeur quelconque du couple (x_1, x_2) dans chaque zone de quantification et nous essayons de classer l'ensemble des valeurs ainsi définies à l'aide d'algorithmes d'apprentissage.



Quantification du domaine de variation de x_1 et x_2 .
(La surface hachurée représente le domaine de bordure)

- FIGURE 16 -

Algorithmes

Les algorithmes que nous utilisons sont les suivants :

- Incrément fixe : $\underline{w}(k+1) = \underline{w}(k) + \frac{\beta}{N} \underline{a}_j^T$ si $\underline{a}_j \cdot \underline{w}(k) < 0$ ($\beta > 0$)
- Relaxation : $\underline{w}(k+1) = \underline{w}(k) + \frac{\pi}{N} (1 - \underline{a}_j \cdot \underline{w}(k)) \underline{a}_j^T$ ($0 < \pi \leq 2$)
- β Variable : $\underline{w}(k+1) = \underline{w}(k) + \beta \frac{\underline{a}_j^T}{N}$

$$\beta = \text{cte} > 0 \quad \text{Si} \quad P_{cc} \leq \Delta < 1$$

$$\beta = \frac{1 \underline{w}(k) 1}{2 P_{cc}^2 N} \quad \text{Si} \quad 1 \geq P_{cc} > \Delta$$

- Moindre carré de l'erreur : $\underline{w}(k+1) = \underline{w}(k) + \pi (\delta - \underline{a}_j \cdot \underline{w}(k)) \underline{a}_j^T$ ($0 < \pi \leq 2$)

δ = valeur minimum admise de l'erreur

Les trois premiers algorithmes s'avèrent converger tandis que le dernier diverge dans certains cas.

Echantillons

Nous considérons, d'une part l'ensemble des échantillons de bordure (zones bordant à droite et à gauche la courbe de commutation $f = 0$) de 38 valeurs, d'autre part l'ensemble des 81 valeurs définies par la quantification (échantillonnage complet).

Enfin, la présentation de chaque couple (x_1, x_2) est faite de deux manières : cyclique et aléatoire. Dans le premier cas, chaque échantillon est présenté à l'élément linéaire dans un ordre déterminé, dans le deuxième cas, les échantillons sont présentés dans un ordre quelconque. Nous avons constaté que le fait de présenter les échantillons au hasard ne modifiait pas les propriétés de convergence des algorithmes. L'expérience a montré une amélioration de la rapidité de la convergence mais, pour des raisons de simplification de calculs, nous avons adopté une présentation cyclique.

Résultats

Nous présentons les résultats de ces essais sur deux graphiques, le premier consacré aux échantillons de bordure (38 éléments), le deuxième à l'échantillonnage complet (81 éléments)

1° - Echantillonnage complet

Les tableaux ci-dessous donnent le facteur de qualité (rapport du nombre d'échantillons correctement classés au nombre total d'échantillons) en fonction du nombre d'adaptations des vecteurs de pondération pour les quatre algorithmes utilisés).

P _{cc}	k
0.877	: 17
0.938	: 40
0.951	: 49
0.963	: 57
0.975	: 62
0.988	: 67
0.975	: 70
1	: 72

P _{cc}	k
0.889	: 20
0.951	: 33
0.926	: 44
0.951	: 51
0.988	: 56
0.988	: 61
0.988	: 63
1	: 64

P _{cc}	k
0.864	: 21
0.877	: 30
0.963	: 39
0.975	: 44
0.963	: 49
1	: 52

P _{cc}	k
0.778	: 22
0.889	: 41
0.951	: 51
0.988	: 55
1	: 56

Incrément fixe

Relaxation

β variable

moindre carré

$$\beta = 3$$

$$\pi = 2$$

$$\Delta = 0.900$$

$$\Pi = 1.500$$

$$\beta = 1.350$$

$$\delta = 0.010$$

Ces résultats sont reportés sur la figure 17.

Deux remarques peuvent être formulées :

- La convergence est assurée par chacun des Algorithmes
- La dispersion des résultats est relativement faible.

On constate également que l'algorithme par moindre carré est celui qui nécessite le moins de cycles de présentation des échantillons (5 cycles, soit un temps de calcul de 70 secondes environ sur IBM 360-25).

2° - Echantillonnage de Bordure

P _{cc}	k
0.684	: 19
0.816	: 37
0.816	: 53
0.842	: 66
0.868	: 78
0.974	: 84
0.974	: 84
1	: 85

P _{cc}	k
0.632	: 21
0.732	: 33
0.789	: 43
0.842	: 52
0.842	: 64
0.842	: 75
0.868	: 83
0.921	: 88
1	: 91

P _{cc}	k
0.684	: 18
0.816	: 34
0.842	: 46
0.868	: 52
0.974	: 66
0.921	: 71
0.974	: 73
1	: 74

P _{cc}	k
0.711	: 26
0.684	: 37
0.789	: 53
0.816	: 72
0.789	: 81
0.816	: 92
0.921	: 100
1	: 111

P _{cc}	k
0.658	: 22
0.684	: 39
0.763	: 61
0.737	: 80
0.711	: 101
0.737	: 122
0.711	: 138
0.711	: 177
0.684	: 214

Incrément fixe

Relaxation

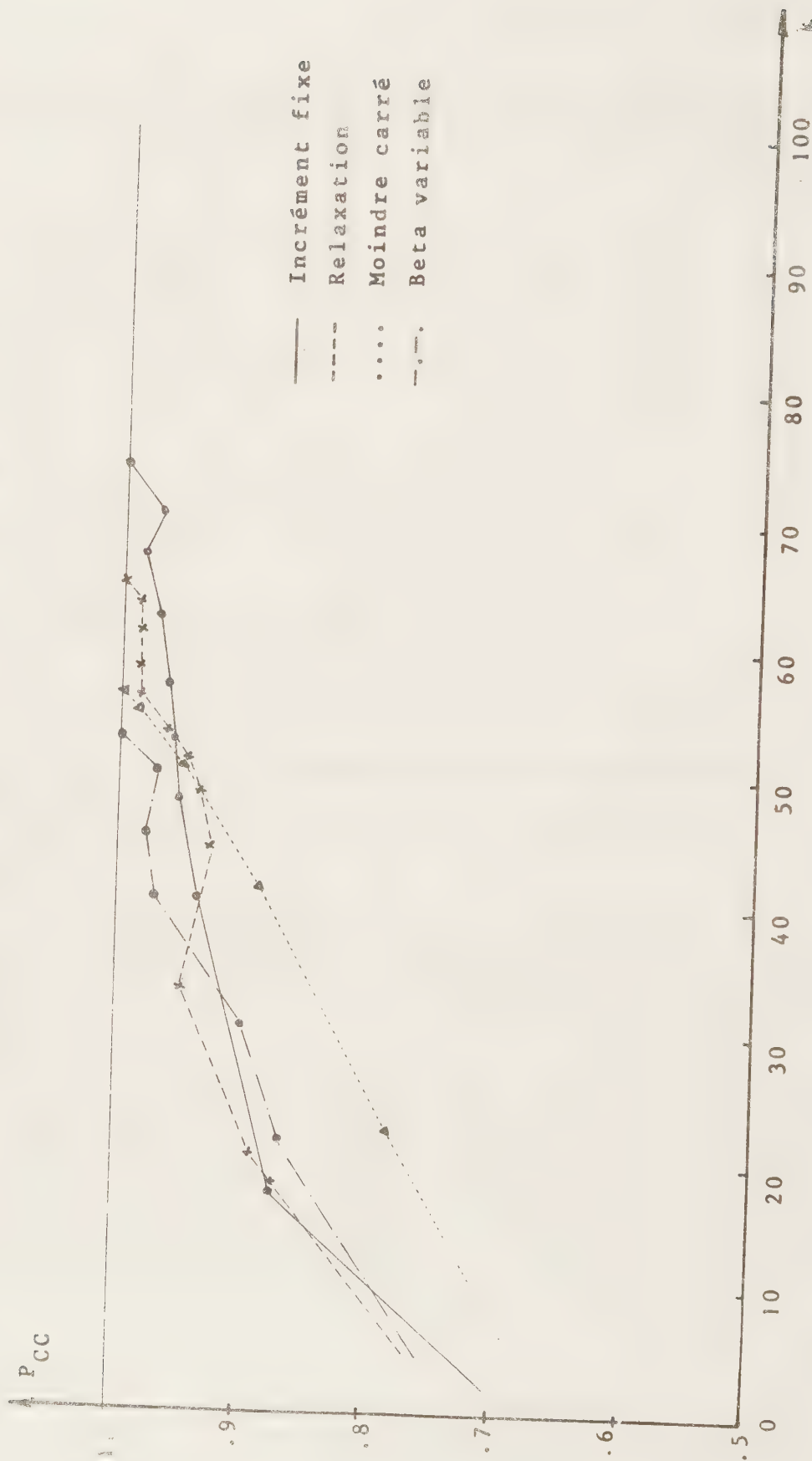
β Variable

Moindre carré

(1)

(2)

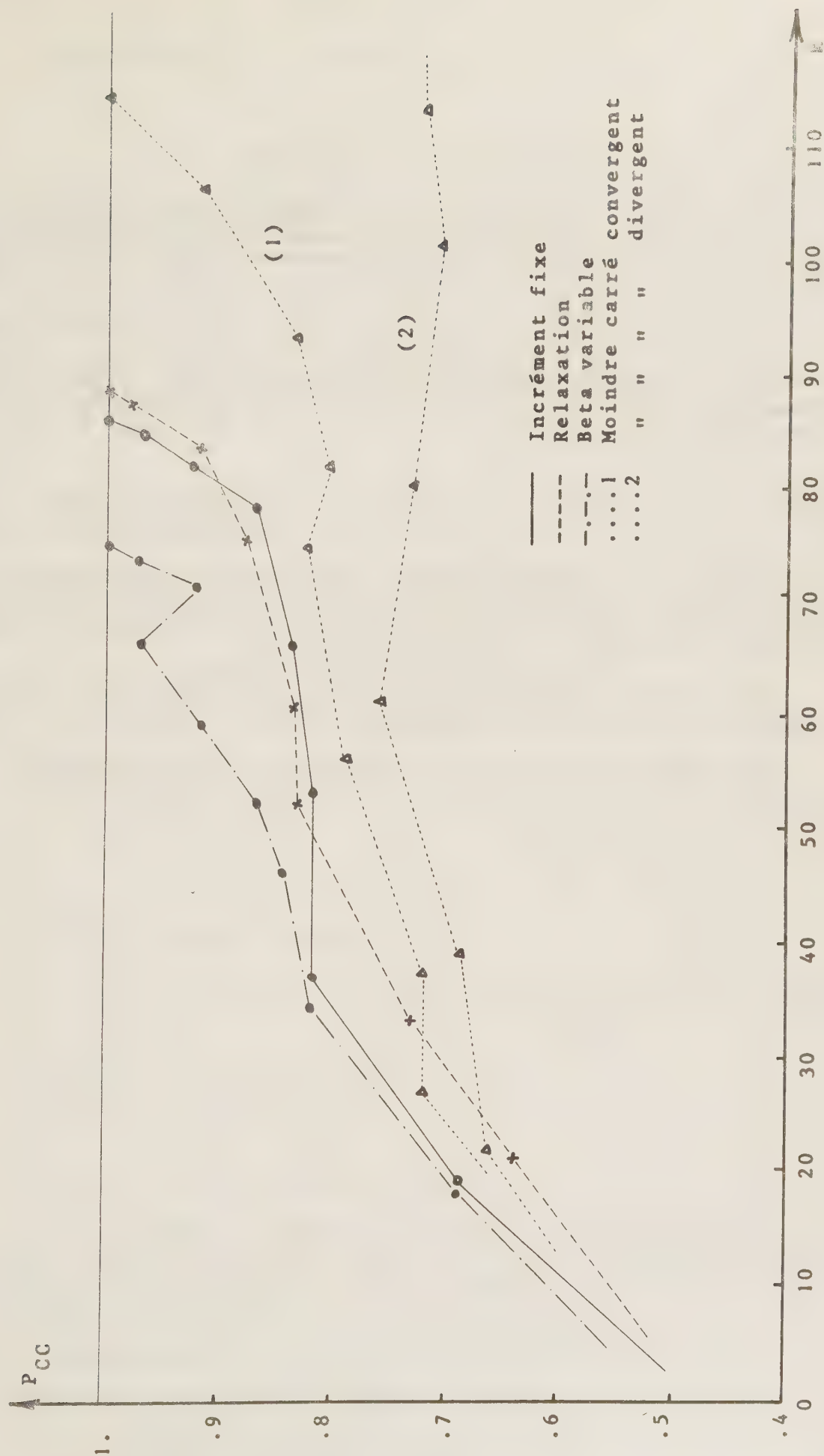
$P_{CC} = \frac{\text{Nombre d'échantillons correctement classés}}{\text{Nombre d'échantillons à classer}}$



Nombre d'adaptations en fonction de l'ALGORITHME utilisé

E C H A N T I L L O N D E B O R D U R E (p r e s e n t a t i o n c y c l i q u e)

-49-



- FIGURE 18 -

Ces résultats sont reportés sur la figure 18.

La convergence est dans ce cas plus longue à obtenir, ce qui correspond à une plus grande difficulté de positionnement de la surface commutation (cf § II - 1 - 3 - C) ; de plus, la dispersion des résultats est plus importante.

Un essai intéressant concerne l'utilisation de l'algorithme du moindre carré de l'erreur. Nous avons représenté sur la figure deux essais caractéristiques avec deux ordres de présentation différents. Le premier ① converge, mais cette convergence est longue à obtenir. Le deuxième ② ne donne pas de résultat et diverge très rapidement.

CONCLUSION

Au vu de ces différents résultats, aussi bien en ce qui concerne l'ensemble des éléments qu'un échantillonnage réduit, deux algorithmes donnent une convergence rapide en un nombre acceptable d'adaptations et de cycles, et sont de mise en oeuvre très simple : Incrément et Relaxation.

Nous retiendrons ces deux algorithmes pour les applications sur processus que nous proposons dans le chapitre III.

Nous donnons en annexe de ce paragraphe un listing de programme FORTRAN utilisé pour ces essais (avec ici l'algorithme à β variable).

APPLICATIONS DE LA COMMANDE SOUS OPTIMALE

Nous avons indiqué, dans l'introduction que la présence de contraintes sur l'Etat rendait la détermination en ligne d'une commande optimale quasiment impossible, mais qu'une stratégie sous-optimale telle que celle décrite au chapitre II paraît bien adaptée à ce genre de problème. C'est là notre motivation de présenter un exemple simple mais non trivial (les équations d'Etat sont aussi bien celles du niveau que celles de la pression dans une chaudière à ballon) où l'un des états est soumis à une contrainte.

Dans une deuxième partie, on montre les possibilités d'applications d'une telle méthode à un processus réel. Enfin, nous donnons en annexe les programmes de calcul qui nous ont servi pour résoudre ces différents problèmes.

III - 1 Etude en simulation de la commande d'un processus en $1/S^2$ (double intégrateur) muni de contraintes.

A partir de l'obtention d'un certain nombre de solutions exactes (trajectoires optimales) de ce problème, nous appliquons la méthode sous-optimale au calcul d'une commande que nous testons en simulation.

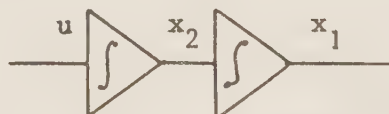
III - 1 - 1 Position du problème d'optimisation

Considérons le système d'équations d'état :

$$\dot{x}_1 = ax_2$$

$$\dot{x}_2 = ku$$

$$|x_2| \leq B$$



Et le critère à minimaliser :

$$J = \frac{1}{2} \int_{t_0}^T (x_1^2 + \alpha^2 u^2) dt$$

T est fixé et t_0 est indifféremment fixé ou le temps courant.

Avec les conditions initiales :

$$x_1(t_0) = x_{1_0}$$

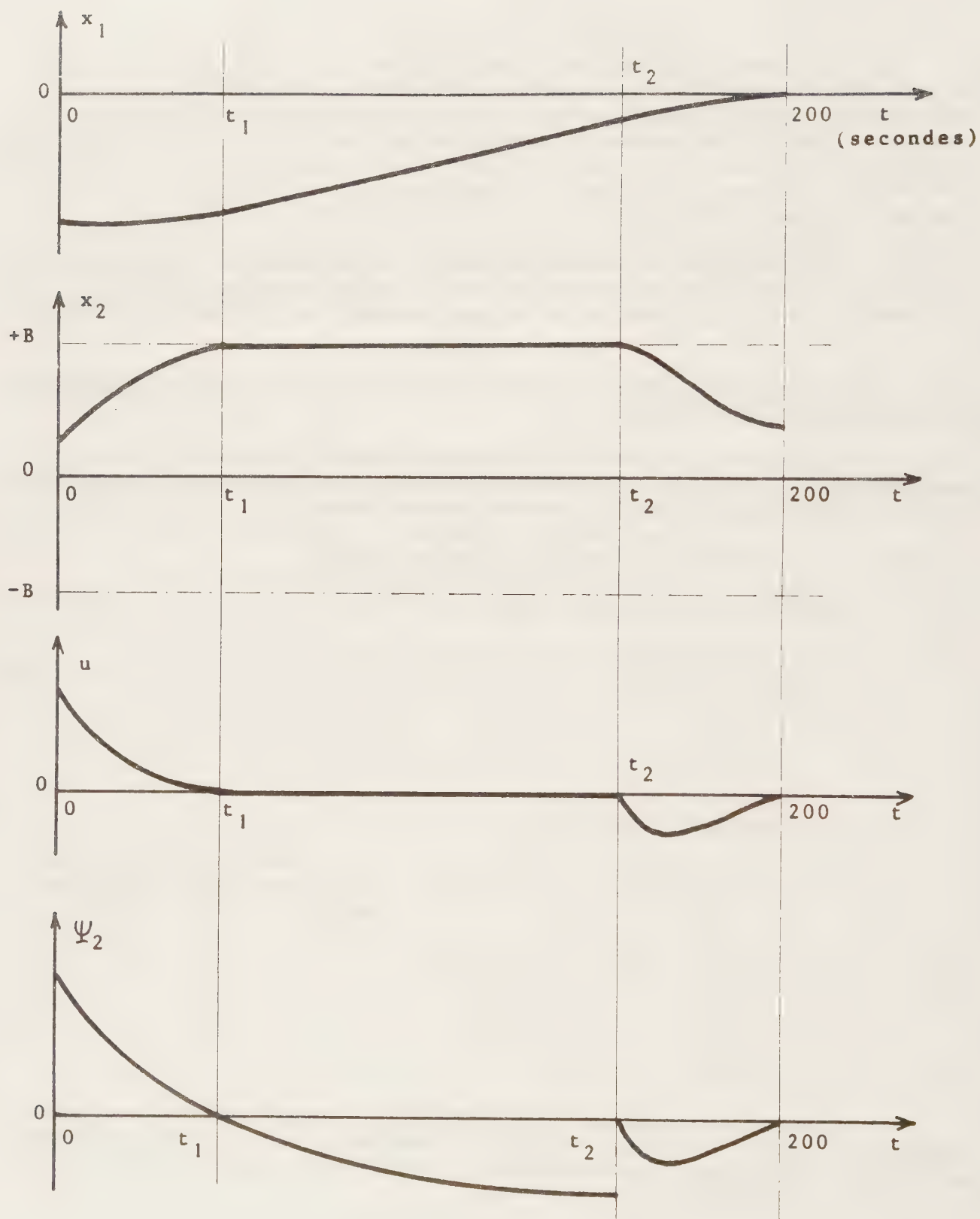
$$x_2(t_0) = x_{2_0}$$

Ces équations traduisent un problème de commande optimale avec contraintes sur l'état (x_2 domaine fermé). La résolution de ce type de problème par les méthodes variationnelles a été obtenue notamment par application du principe du maximum de PONTYAGIN (1).

Nous donnons ici les conditions d'optimalité à remplir :

Hors contraintes, la commande u minimise le Hamiltonien :

$$H = \frac{1}{2} (x_1^2 + \alpha^2 u^2) + \psi_1 ax_2 + \psi_2 ku$$



-FIGURE 19 -

Ce qui fournit $u = - \frac{k}{\alpha_2} \psi_2$

Avec : $\dot{\psi}_1 = - \frac{\partial H}{\partial x_1} = - \alpha_1$ et $\psi_1(T) = \psi_2(T) = 0$

$$\dot{\psi}_2 = - \frac{\partial H}{\partial x_2} = - \alpha \psi_1$$

En contrainte, on a $x_2 = \pm B$, donc $\dot{x}_2$ doit être nul, soit $u = 0$

D'autre part, on montre (1) qu'aux points d'entrée et de sortie de butée, l'Hamiltonien est continu, ce qui implique, à l'instant où on entre en contrainte :

$$\frac{1}{2} [(x_1^-)^2 - \alpha^2 (u^-)^2] + \psi_1^- \alpha x_2^- = \frac{1}{2} (x_1^+)^2 + \psi_1^+ \alpha x_2^+$$

Comme ψ_1 , x_1 et x_2 sont continus, on a $u^- = 0$ ce qui donne $\psi_2^- = 0$ et signifie que u est continu sur l'intervalle $[t_0, T]$.

Finalement, les trajectoires types (plausibles physiquement) sont celles données sur la figure 19.

La détermination des instants t_1 et t_2 d'entrée et de sortie de contraintes constitue la principale difficulté par rapport aux problèmes classiques d'optimisation sans contrainte.

Un algorithme de résolution du problème à valeurs limites en plusieurs points (conditions limites données aux instants t_0 , t_1 , t_2 et T) a été mis au point par Monsieur ABDALLAH (2) et nous a permis d'obtenir l'échantillon de trajectoires optimales exactes dont nous avons besoin.

La commande sous-optimale à élaborer doit ici se présenter sous la forme $u = h(x_1(t), x_2(t), T-t, \alpha)$ (ici t_0 est devenu le temps courant) où α et T sont constants et sont à la disposition de l'utilisateur qui peut ainsi choisir les réponses lui paraissant "les meilleures". Dans la suite cependant, on a admis que α est fixé et que T est ∞ : Cette hypothèse ne restreignant en rien la validité de l'exemple vis à vis du problème de contrainte.

III - 1 - 2 Choix d'un échantillon de solutions exactes :

Reprenons les équations d'état :

$$\begin{aligned} \dot{x}_1 &= 1,25 x_2 & \alpha &= 10 \\ \dot{x}_2 &= 6 \cdot 10^{-3} u & T &= \infty \\ \text{(Avec } j &= \frac{1}{2} \int_t^\infty (x_1^2 + \alpha^2 u^2) dt \end{aligned}$$

(Notons que du point de vue du calcul des trajectoires ∞ peut être remplacé par 200 secondes).
 $|x_2| < B = 0.020$

valeurs numériques se réfèrent à la chaîne de pression de notre unité

Trajectoires optimales (fig 20) calculées à partir de ces équations

sont symétriques par rapport à l'origine du plan x_1, x_2 . Les butées x_2 d'une part, les limites des valeurs plausibles de x_1 d'autre part font que x_1 et x_2 évoluent à l'intérieur d'un domaine fermé défini comme suit :

En x_1 : - 3 , + 3
En x_2 : - B ; + B

Les limites de variations de x_1 ont été prises par référence aux écarts de pression admissibles sur notre pilote.

Ces différentes considérations nous ont amenés à choisir un échantillon de 22 trajectoires, réparti sensiblement uniformément comme le montre la figure 20 (Le trait continu traduit une commande positive, le trait pointillé une commande négative). On peut remarquer que le lieu des points de changement de signe (également des "sommets" des trajectoires) est sensiblement une droite passant par l'origine).

III - 1 - 3 Quantification - Choix d'un Code

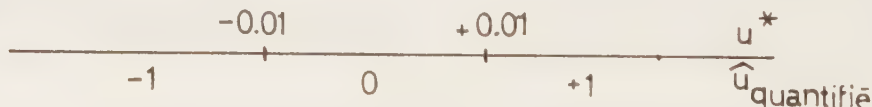
A - Quantification de la commande

La commande sous-optimale préconisée n'a de valeur que si elle reste suffisamment simple à mettre en oeuvre : or cette simplicité est liée étroitement au nombre de zones de quantification des variables x et u et plus particulièrement de u . Ici u est une accélération ou plus exactement la dérivée d'un débit : il est donc possible de ne prendre pour la commande que trois valeurs +1, -1, ou 0, qui correspondent à une augmentation, une descente ou à un maintien du débit de combustible.

Le problème réside maintenant dans le choix de zones de commande correspondant à ces trois valeurs +1, -1 et 0 : Nous avons ici considéré comme zone +1, Les commandes supérieures à la valeur moyenne m des commandes montrées dans l'échantillon. Notons qu'il est souvent nécessaire de descendre à $\frac{m}{2}$.

La valeur de u analogique appliquée ensuite sur le processus étant prise égale à cette valeur moyenne.

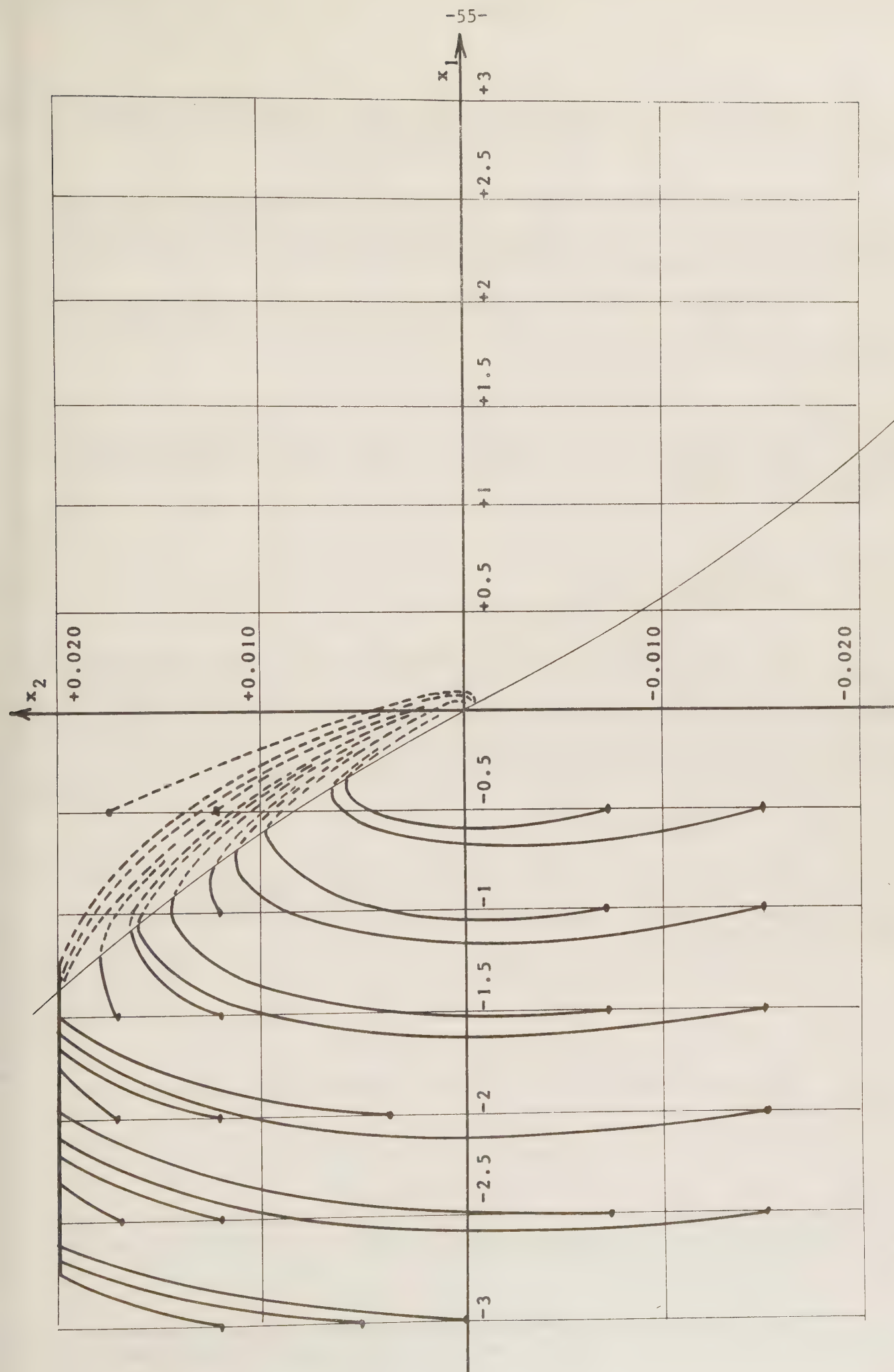
Ces deux choix sont garants de l'obtention des trajectoires voisines des trajectoires optimales. Nous avons pris comme limites des zones +1, -1 et 0 : ± 0.01 , ce qui se traduit sur la figure 21 par deux courbes de séparation AA' et BB' sensiblement linéaires, qui constituent les limites des catégories +1, -1 et 0.



B - Quantification de l'Etat

Nous avons quantifié le domaine de variation de (x_1 et x_2) suivant la figure 21.

En x_2 : 8 intervalles égaux (0.005) entre butées, plus deux intervalles ouverts au delà de chaque niveau de butée. Ces dix intervalles sont répertoriés de 15 à 24.



- FIGURE 20 -

En x_1 : 12 intervalles égaux (0.25) entre - 1,5 et + 1,5, plus deux intervalles ouverts au delà de ces deux valeurs; ces intervalles sont répertoriés de 1 à 14.

La quantification serrée à l'intérieur du domaine (-1,5 , +1,5) est due au fait que x_1 représente un écart en pression (bars) et que la précision des capteurs de pression est de l'ordre de 1% de l'échelle totale de mesure, ce qui pour notre pilote, représente 1% de 30 bars, soit $\approx 0,3$ bars.

A l'extérieur de ce domaine, on peut admettre que , étant éloigné du but (0), la commande ne s'annule jamais (sauf si $|x_2| = B$) et donc est indépendante du point où l'on se trouve.

Compte tenu de la quantification de l'Etat, ces deux courbes de commutation sont approximées par deux courbes en escalier, telles qu'elles sont dessinées sur la figure.

C - Choix d'un code

Nous avons choisi un code mono-élément de part sa simplicité de mise en oeuvre sur calculateur numérique. De plus, ce code est linéairement indépendant. Enfin comme le montre plus loin le listing du programme FORTRAN d'apprentissage hors ligne, ce code permet d'utiliser une méthode d'adaptation qui simplifie à l'extrême le calcul de chaque produit $\underline{a}_j^T \underline{w}(k)$. En effet, ce terme est égal à la somme de deux composantes de $\underline{w}(k)$. Par exemple, si $\underline{a}_j^T$ représente le vecteur ligne quantifié de $x_1 \in \text{zone } 6$ et $x_2 \in \text{zone } 19$, le produit $\underline{a}_j^T \underline{w}(k) = w_k(6) + w_k(19)$

III - 1 - 4 Apprentissage

Revenons à la figure 2 dans le plan des phases. La forme à apprendre est quelque peu compliquée, en raison de la présence des contraintes, ce qui posera quelques difficultés de convergence et de précision à l'algorithme de reconnaissance de forme.

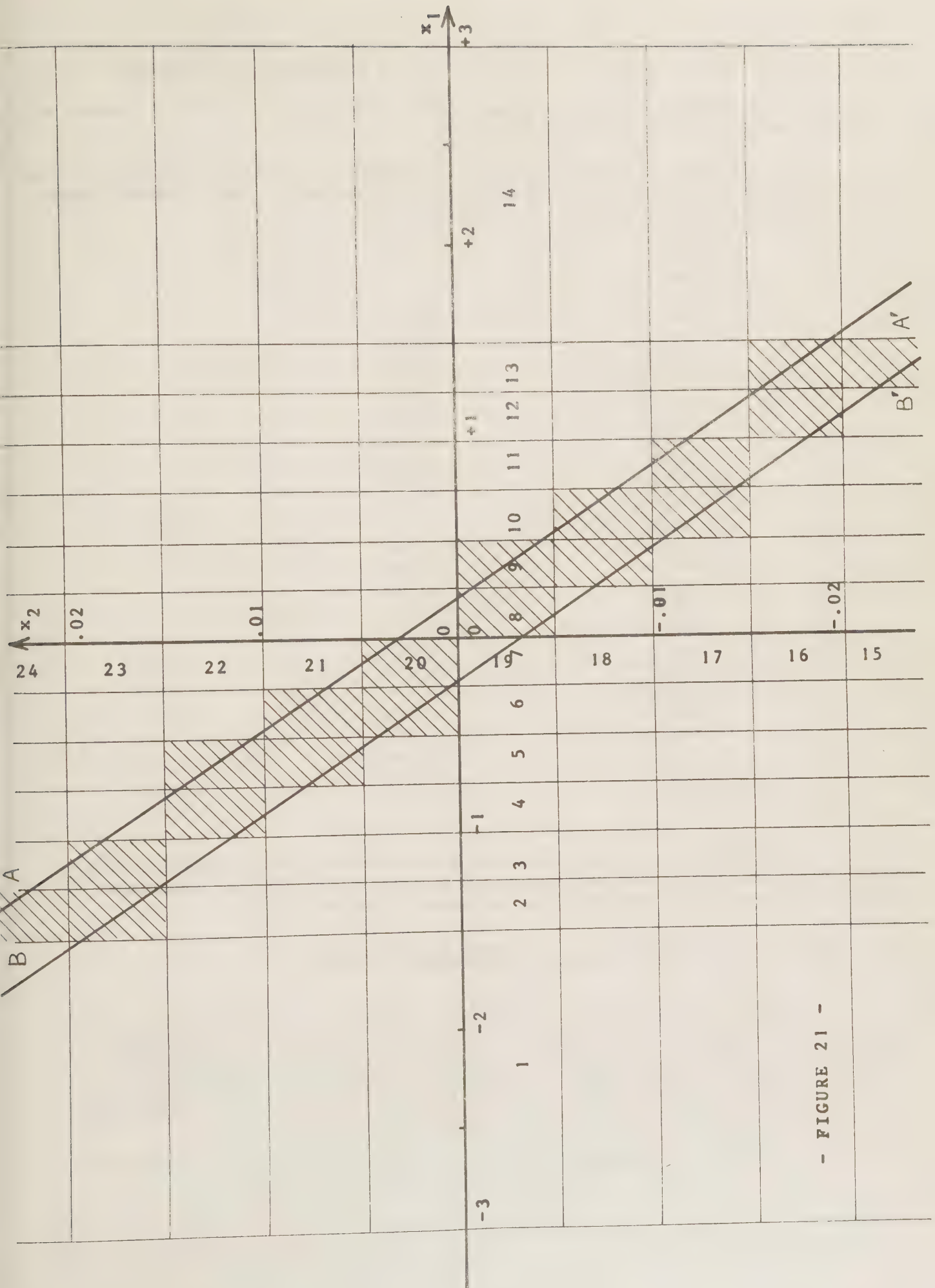


On a donc considéré que la forme à apprendre était celle de la figure ci-dessous



et la prise en considération des contraintes est effectuée numériquement à l'aide de la logique : ainsi dans le cas où $x_1 < 0$, on a, si $x_2 = B$ et $u_1 = +1$ ou 0 , appliquer $u = 0$, si $u_1 = -1$ $u = -1$.

Les échantillons sont présentés dans un ordre cyclique. L'apprentissage est effectué compte tenu des contraintes $\pm B$. Ceci permet une amélioration du



- FIGURE 21 -

temps de calcul hors-ligne des vecteurs de pondération, et n'enlève rien au principe de la méthode, dans la mesure où les courbes de commutation sont déterminées à partir de solutions optimales exactes en présence de contraintes.

La détermination des vecteurs poids se fait selon la méthode exposée en II-2, à l'aide de l'algorithme de relaxation.

Le tableau ci-dessous donne les vecteurs de pondération, tandis que nous portons sur la figure 22 les courbes de convergence, de cette phase d'apprentissage correspondant au calcul des deux vecteurs WA et WB.

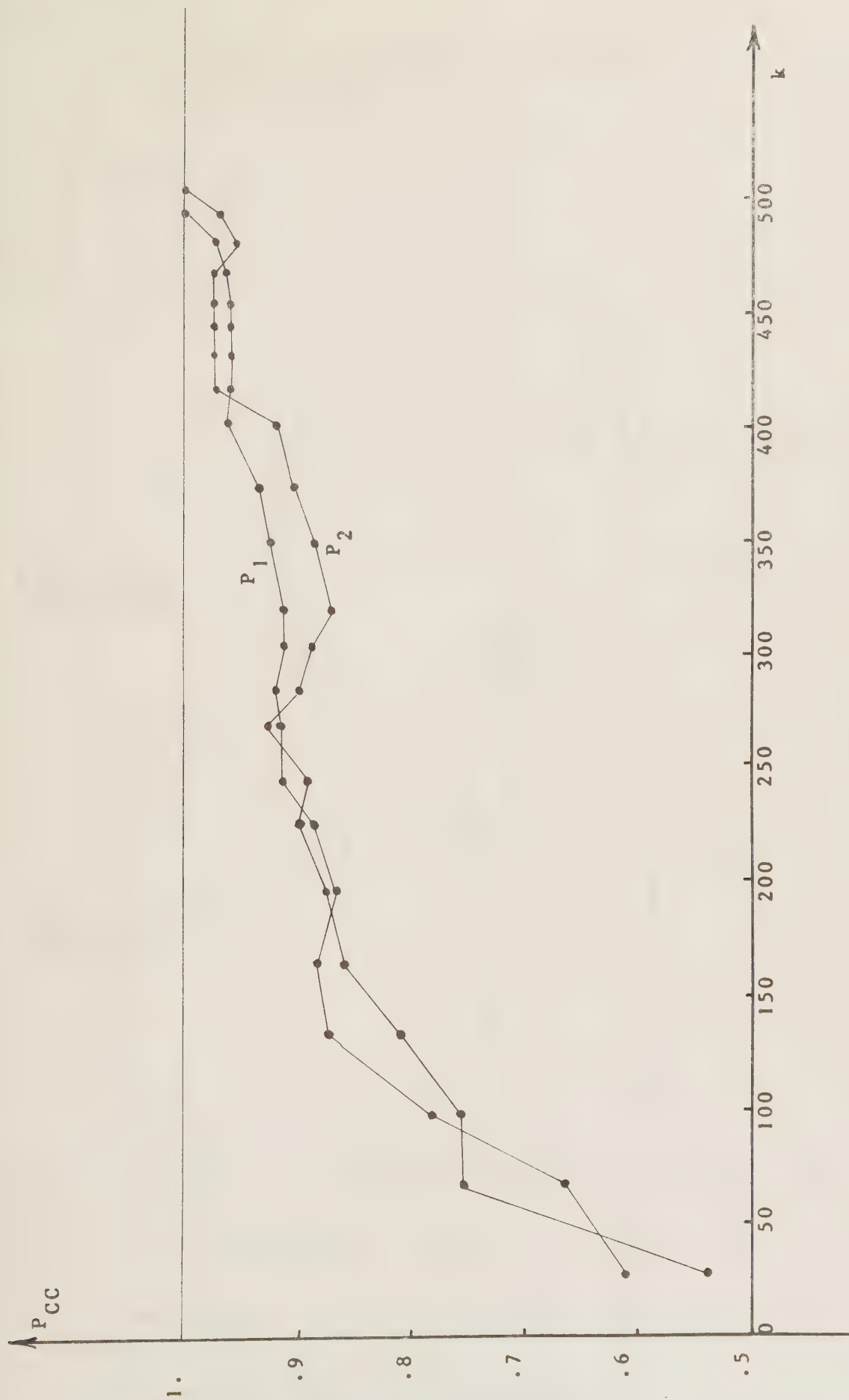
WA	WB
+0.513	+0.514
+0.394	+0.514
+0.402	+0.395
+0.289	+0.289
+0.0184	+0.302
+0.079	+0.184
+0.095	+0.078
-0.024	-0.023
-0.129	-0.007
-0.233	-0.130
-0.221	-0.234
-0.444	-0.340
-0.549	-0.328
-0.541	-0.550
+0.495	+0.499
+0.286	+0.405
+0.180	+0.286
+0.077	+0.183
-0.027	+0.078
-0.132	-0.026
-0.236	-0.131
-0.342	-0.238
-0.450	-0.341
-0.437	-0.452

Le calcul numérique hors-ligne a été mené sur IBM 1130 (mémoire 8k); le temps de calcul des deux vecteurs est d'environ 70 secondes. Le programme FORTRAN de calcul simultané est donné en annexe de ce chapitre.

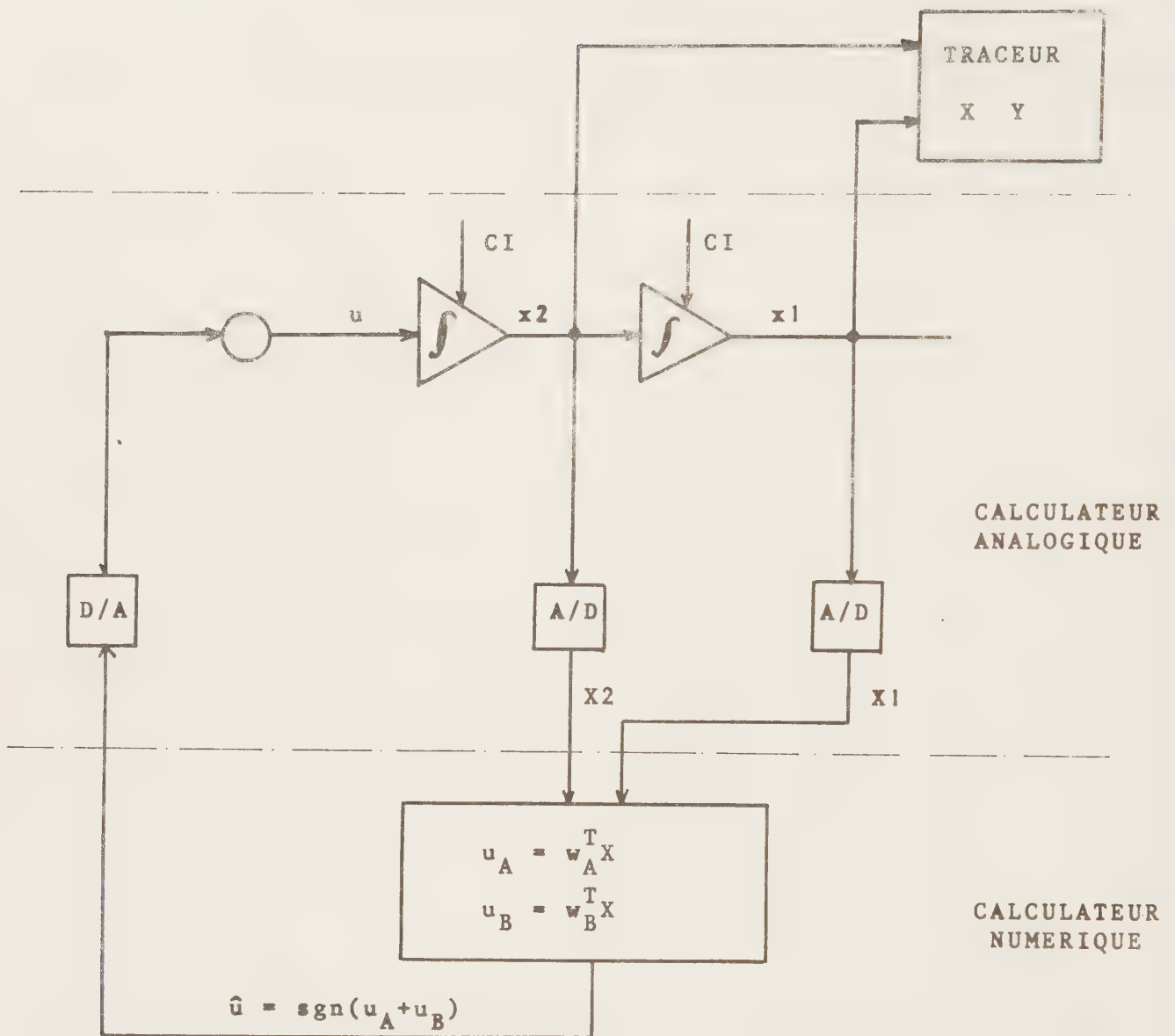
III - 1 - 5 Simulation de la commande sous-optimale

Nous effectuons cette simulation de la manière suivante. Les équations d'Etat sont câblées sur la calculateur analogique TR 48 (EAI),, les grandeurs analogiques x_1 , x_2 sont converties en grandeurs numériques et traitées dans un calculateur EAI 640, qui calcule la commande correspondante. Cette commande est convertie en grandeur analogique, pondérée, et appliquée au système analogique. Les Etats x_1 et x_2 sont enregistrés sur traceur XY.

La figure 23 schématise cette installation hybride.



- FIGURE 22 -



ENSEMBLE HYBRIDE DE SIMULATION

A - Mise en oeuvre du calculateur analogique

Reprenons les équations d'Etat :

$$\dot{x}_1 = 1,25 x_2$$

$$\dot{x}_2 = ku$$

Nous choisissons les calibres suivants :

$$x_1 \text{ cal} = 5 \text{ bars}$$

$$x_2 \text{ cal} = 0.05 \text{ bar/s}$$

$$\dot{x}_1 \text{ cal} = 0.1 \text{ bar/s}$$

$$\dot{x}_2 \text{ cal} = 0.1 \text{ bar/s}^2$$

D'où les Equations "Machine"

$$\left(\frac{\dot{x}_1}{0.1} \right) = 1,25 \frac{1}{2} \left(\frac{x_2}{0.05} \right) = 0,625 \left(\frac{x_2}{0.05} \right)$$

$$\left(\frac{\dot{x}_2}{0.1} \right) = \left(\frac{u}{0.1} \right)$$

Calibrage des equations de fonctionnement des intégrateurs :

$$x_1 = \int \dot{x}_1 dt + x_1(0) \quad x_1(0) = x_{10}$$

$$x_2 = \int \dot{x}_2 dt + x_2(0) \quad x_2(0) = x_{20}$$

Soit :

$$\left(\frac{x_1}{5} \right) = 0,02 \int \left(\frac{\dot{x}_1}{0,1} \right) dt + \frac{x_{10}}{5}$$

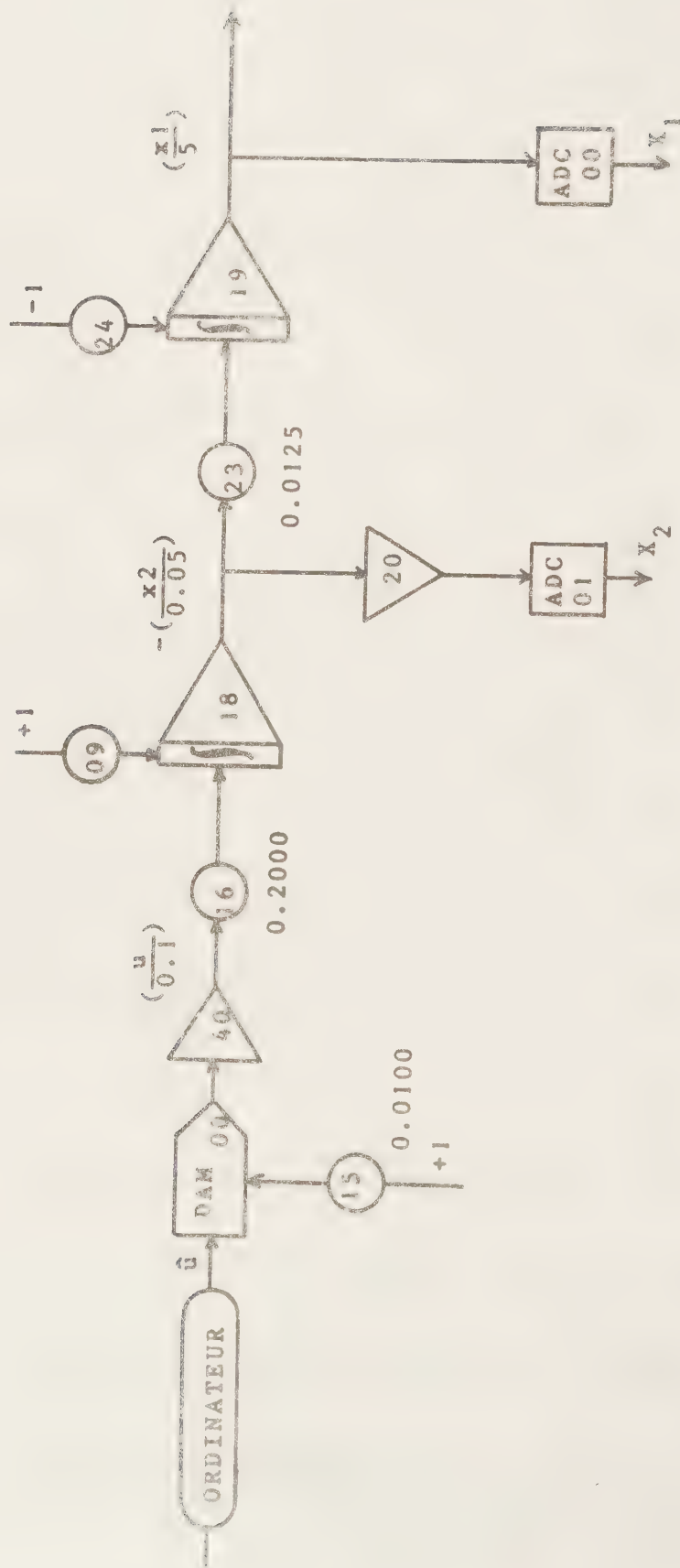
$$\left(\frac{x_2}{0,05} \right) = 2 \int \left(\frac{\dot{x}_2}{0,1} \right) dt + \frac{x_{20}}{0,05}$$

D'où le schéma de câblage : figure 24

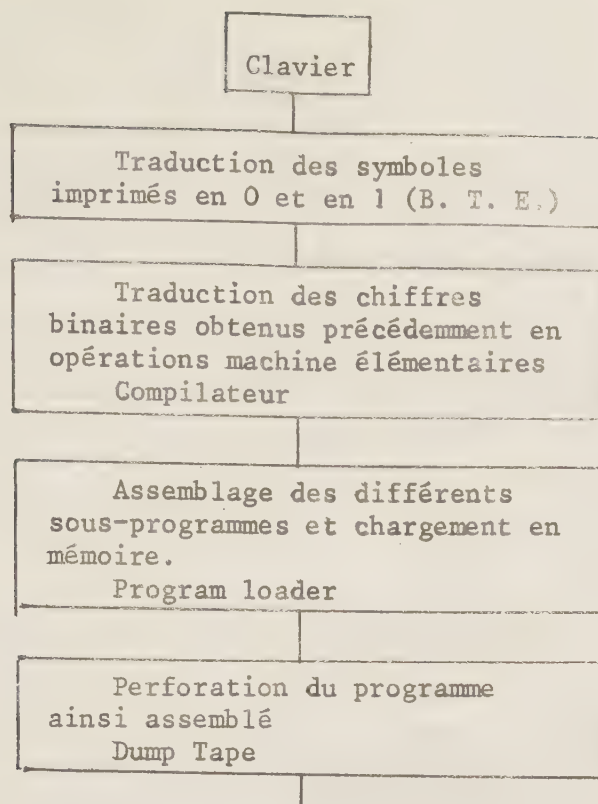
B - Mise en oeuvre du calculateur numérique

Nous avons adopté le langage FORTRAN IV pour établir l'acquisition des données x_1 et x_2 et t , et le calcul de la commande sous-optimale u .

Les entrées sorties du calculateur EAI 640 se font par bandes perforées, avec accès en entrée au clavier et contrôle sur télétype. La capacité de mémoire étant à l'heure actuelle relativement modeste, l'utilisateur ne dispose pas de super-viseur et doit assembler les éléments de son programme. Cet assemblage se fait de la manière indiquée sur l'organigramme ci-après.



Câblage analogique (TR 48)



Le calcul de la commande u à partir des informations données sur l'Etat x_1 et x_2 fait appel aux vecteurs de pondération WA et WB , qu'il faut donc mettre en mémoire dans le calculateur.

Le programme d'acquisition de données se batit d'une manière identique à celle utilisée pour l'élaboration du programme principal. (le listing FORTRAN est donné en annexe). Une bande "DONNEES" est alors élaborée.

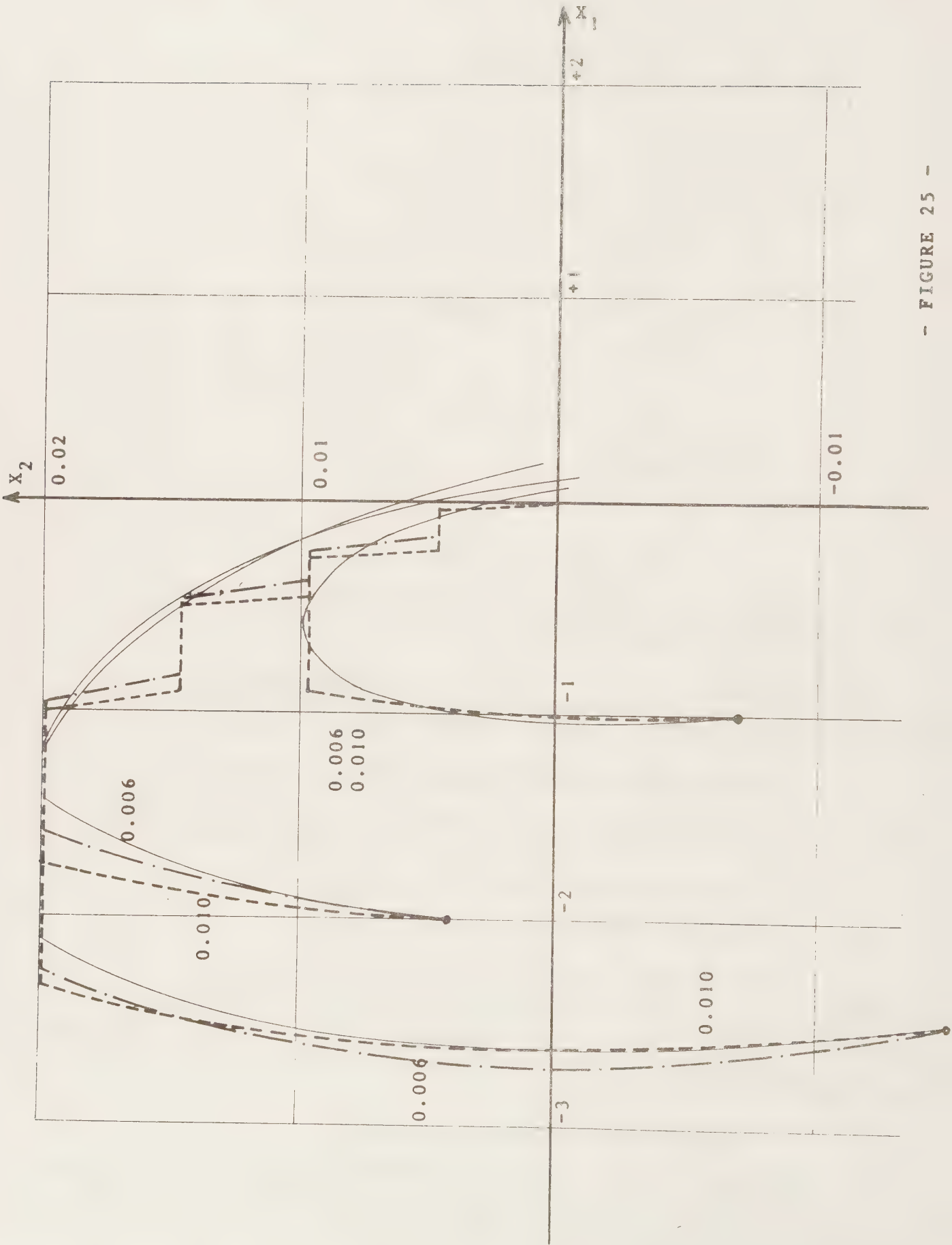
C - Mise en Route du PROGRAMME de calcul de la commande u

- Mettre en mémoire le programme principal (Bande "PROGRAMME PRINCIPAL C.I., adresse '17777) à l'aide du C.I.L. (core Image Loader).
- Placer la bande " DONNEE" sur le lecteur.
- Exécuter le programme en '1000.

La bande de données est lue et le programme se déroule en continu, c'est-à-dire qu'il est bouclé sur lui-même, et recommence à travailler après chaque calcul d'une commande. Signalons que ce calcul dure quelques 1/10' de seconde, et que le calcul proprement dit de la commande, compte non tenu de la quantification des états, demande 10 instructions FORTRAN (5 pour chaque courbe de commutation).

III - 1 - 6 Résultats expérimentaux et Conclusion

Nous avons choisi de comparer les commandes optimales et sous optimales pour trois trajectoires types : une trajectoire sans contrainte (conditions initiales $x_{10} = -1$, $x_{20} = 0.007$); Deux trajectoires avec contrainte positive, la première étant initialement peu éloignée du but ($x_{10} = -2$, $x_{20} = 0.004$) la deuxième étant très éloignée ($x_{10} = -2,5$, $x_{20} = -0,015$)



- FIGURE 25 -

Les trajectoires optimales, calculées d'après (3), sont préalablement traçées sur la feuille du VARILOTTER. Le câblage analogique est alors initialisé à chacune des trois valeurs initiales et le programme numérique mis en fonctionnement, puis le calculateur analogique est mis en mode commande. Chaque trajectoire sous optimale est alors tracée en superposition sur les trajectoires optimales correspondantes.

Nous avons effectué ces essais pour deux valeurs de pondération de la commande sous optimale : $u=0.01$ (valeur de quantification) et $u=0.006$ (valeur moyenne des commandes optimales en sortie de butée).

Les trajectoires résultantes sont données sur la figure 25.

Les résultats obtenus montrent, pour ce problème, l'intérêt de la méthode de reconnaissance de formes. En effet les trajectoires obtenues sont très proches des trajectoires optimales et ceci malgré la simplification extrême de la commande (+1, 0, -1).

III - 2 APPLICATION de la méthode d'apprentissage à un problème réel.

Partant des équations du modèle mathématique établi au chapitre I, nous donnons une méthode de calcul des trajectoires optimales. Nous testons alors en simulation les commandes résultantes. Nous appliquons ensuite hors-ligne la méthode d'apprentissage, en insistant sur les problèmes de quantification et de choix des échantillons. Les commandes sous-optimales obtenues par reconnaissance de forme sont essayées en ligne avec le modèle simulé, puis appliquées au processus réel (chaudière pilote).

III-2-1 Position du problème

Nous avons testé la méthode d'apprentissage sur le générateur de vapeur du Laboratoire d'Automatique, décrit au chapitre I, en prenant pour commandes u les dérivées des débits de combustible et d'eau de désurchauffe.

En supposant que le générateur de vapeur fonctionne en télé réglage avec une charge moyenne de 1000 Kg/h, nous avons obtenu des commandes optimales exactes par la théorie de KALMAN (3).

Rappelons les équations du modèle mathématique (cf. Chapitre I) :

$$\begin{aligned} \dot{p}_B &= 0.00034q_c - 0.0000281(q_s - q_d) \\ 150\dot{t}_s &= -t_s + 0.8(q_c - 80) - 0.1(q_s - q_{sN}) - z \\ 50\dot{z} &= -z + 0.8(q_d - 40) \\ \dot{q}_c &= u_1 \\ \dot{q}_d &= u_2 \end{aligned} \quad \left(\text{III} - 1 \right) \quad \left[\begin{aligned} q_{sN} &= 1000 \text{ kg/h} \\ \delta^2 &= \epsilon^2 = 30 \\ \alpha^2 &= 16 \\ \delta q_s &= q_s - q_{sN} \\ \delta q_s &= \text{bruit blanc, pour } q_s \\ &\quad \text{voisin de } q_{sN} \end{aligned} \right.$$

Avec :

$$P_S = P_B - 1.07 \left(\frac{q_s}{q_{sN}} \right)^2 \approx P_B - 1.07 \left(1 + 2 \frac{\delta q_s}{q_{sN}} \right)$$

Le critère étant :

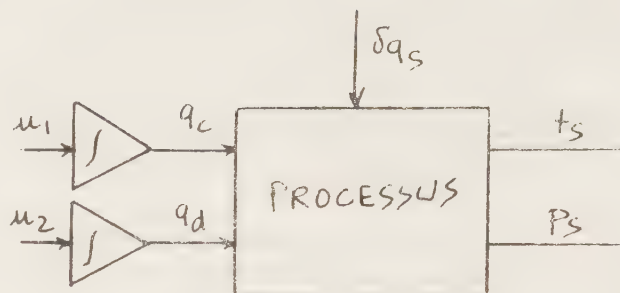
$$J = E \left\{ \frac{1}{2} \int_t^{t+T} \left(t_s^2 + \alpha^2 p_s^2 + \delta^2 \dot{q}_c^2 + \epsilon^2 \dot{q}_d^2 \right) dt / \text{mesures} \right\}$$

E : espérance mathématique

Ici, T, α , δ et ϵ sont imposés, de même que q_{sN} . Sont considérés comme variables d'état :

p_B , q_c , t_s , z et q_d .

On remarque que, compte tenu de la figure 2 (chapitre I), la perturbation q_s a été assimilée à un processus de Wiener, ce qui signifie qu'en supposant tous les états exactement mesurables, le problème peut être considéré comme déterministe, si l'on prend :



$\delta q_s(\tau) = \delta q_s(t) \quad \tau \in \{t, t+T\}$ (5), le critère étant alors :

$$J = \frac{1}{2} \int_t^{t+T} (t_s^2 + \alpha^2 p_s^2 + \delta^2 \dot{q}_c^2 + \epsilon^2 \dot{q}_d^2) dt$$

III-2-2 Calcul des trajectoires optimales

Nous exposons ici une méthode donnant le vecteur de commande $\underline{u}$ en fonction du vecteur d'état $\underline{X}$ (problème bien connu du "régulateur").

Le vecteur d'état $\underline{X}$ est égal à $\underline{X} = \begin{bmatrix} p_B \\ q_c \\ t_s \\ z \\ q_d \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}$

Le vecteur de sortie $\underline{y} = \begin{bmatrix} p_s \\ t_s \end{bmatrix}$

Le vecteur de commande $\underline{u}$ est égal à $\underline{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} \dot{q}_c \\ \dot{q}_d \end{bmatrix}$

et les équations d'état et de sortie :

$$\dot{\underline{X}} = \underline{A}\underline{X} + \underline{B}\underline{u} + \underline{\chi} \quad (\text{III-2})$$

$$\underline{y} = \underline{C}\underline{X} + \underline{\xi}, \text{ en supposant pour l'instant que } \delta q_s = 0$$

Le principe du maximum conduit au calcul du Hamiltonien H :

$$H = -\frac{1}{2} \left\{ x_3^2 + \alpha^2 \left(x_1 - 1.07 \left[\frac{q_s}{q_{sN}} \right]^2 \right)^2 + \delta^2 u_1^2 + \epsilon^2 u_2^2 \right\} + \underline{\psi}^T (\underline{A}\underline{X} + \underline{B}\underline{u} + \underline{\chi})$$

$\underline{\psi}$ étant le vecteur adjoint, de composantes $\psi_1, \psi_2, \psi_3, \psi_4$ et ψ_5 .

Le vecteur de commande est calculé en écrivant que $H_{\underline{u}} = 0$, soit :

$$\underline{u}^* = \begin{bmatrix} \frac{\psi_2}{\delta^2} \\ \frac{\psi_5}{\epsilon^2} \end{bmatrix}$$

En reportant $\underline{u}^*$ dans l'Equation d'état:

$$\dot{\underline{X}} = \underline{A}\underline{X} + \underline{B}\underline{F}\underline{\Psi} + \underline{\chi} \quad \text{avec} \quad \underline{F} = \begin{pmatrix} 0 & 1/\delta^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/\epsilon^2 \end{pmatrix} \quad (\text{III-3})$$

Le vecteur adjoint est déterminé en écrivant que:

$$\dot{\underline{\Psi}} = -\underline{H}_{\underline{X}}$$

On obtient alors:

$$\dot{\underline{\Psi}} = -\underline{A}^T \underline{\Psi} + \underline{D}\underline{X} + \underline{g} \quad (\text{III-4})$$

Supposons qu'il existe une relation linéaire entre $\underline{\Psi}$ et $\underline{X}$, de la forme:

$$\underline{\Psi} = \underline{K}\underline{X} - \underline{\phi} \quad (\text{III-5})$$

En dérivant l'équation (III-5) et en tenant compte de (III-3) et (III-4) on obtient :

$$\dot{\underline{\Psi}} = \dot{\underline{K}}\underline{X} + \underline{K}\dot{\underline{X}} - \dot{\underline{\phi}}$$

puis :

$$-\underline{A}^T \underline{\Psi} + \underline{D}\underline{X} + \underline{g} = \dot{\underline{K}}\underline{X} + \underline{K}(\underline{A}\underline{X} + \underline{B}\underline{F}\underline{\Psi} + \underline{\chi}) - \dot{\underline{\phi}}$$

soit :

$$[\dot{\underline{K}} + \underline{A}^T \underline{K} + \underline{K}\underline{A} + \underline{K}\underline{B}\underline{F}\underline{K} - \underline{D}] \underline{X} - [\dot{\underline{\phi}} + \underline{A}^T \underline{\phi} + \underline{K}\underline{B}\underline{F}\underline{\phi} - \underline{K}\underline{\chi} + \underline{g}] = 0$$

Cette relation devant être vérifiée quelque soit $\underline{X}$, on obtient :

$$\dot{\underline{K}} + \underline{A}^T \underline{K} + \underline{K}\underline{A} + \underline{K}\underline{B}\underline{F}\underline{K} - \underline{D} = 0$$

$$\dot{\underline{\phi}} + (\underline{K}\underline{B}\underline{F} + \underline{A}^T) \underline{\phi} + \underline{g} - \underline{K}\underline{\chi} = 0 \quad (\text{III-6})$$

On peut montrer que $\underline{K}$ est une matrice symétrique définie positive.

De plus les conditions de transversalité imposent les valeurs limites :

$$\begin{aligned} \underline{K}(T) &= (0) \\ \underline{\phi}(T) &= 0 \end{aligned} \quad (\text{III-7})$$

Ces équations étant intégrables à rebours de T à 0.

Les équations (III-6) et (III-7) permettent de calculer la matrice $\underline{K}$ et le vecteur $\underline{\phi}$ en régime permanent (c'est-à-dire $T = \infty$), la convergence des résultats étant assurée pour $T = 600$ secondes. Le programme FORTRAN de calcul numérique de $\underline{K}$ et $\underline{\phi}$ a été mis au point au Laboratoire d'Automatique (4). Ce programme fait appel à un sous programme d'Intégration (H.P.C.G.) fourni par le software I.B.M. (Librairie FORTRAN).

Nous donnons ci-dessous les résultats obtenus pour $\underline{K}$ et $\underline{\phi}$, avec les coefficients des équations d'Etat mentionnés plus haut. Le programme d'Intégration est donné en annexe de cette étude. (temps de calcul : 8 minutes sur I.B.M. 360-40, 40 minutes sur I.B.M. 360-25).

$$\underline{\phi} = \begin{bmatrix} -0.37476\text{E } 04 \\ -0.96467\text{E } 02 \\ -0.15022\text{E } 03 \\ 0.14507\text{E } 02 \\ -0.29161\text{E } 02 \end{bmatrix}$$

$$K = \begin{bmatrix} -0.19268E 04 & -0.12372E 02 & 0.58655E 02 & -0.17381E 02 & -0.18077E 02 \\ & -0.11543E 01 & -0.35291E 01 & 0.67516E 00 & 0.22196E 00 \\ & & -0.36061E 02 & 0.48520E 01 & 0.17660E 01 \\ & & & -0.99808E 00 & -0.53471E 00 \\ & & & & -0.70323E 00 \end{bmatrix}$$

A partir des valeurs de K et ϕ , on obtient la commande $\underline{u}$ en écrivant que :

Soit :
$$\underline{u} = F \underline{\psi}$$
$$\underline{u} = FK\underline{X} - F\phi$$

et
$$\underline{u}(t) = FK(t) \underline{X}(t) - F\phi(t) \quad (III-8)$$

Cette relation donne le vecteur commande optimal correspondant à des conditions initiales $\underline{X}(0)$

Si maintenant δq_s est $\neq 0$, $\underline{u}$ s'écrit :

$$\underline{u}(t) = FK(t) \underline{X}(t) - F\phi(t) - \gamma \delta q_s \quad (III-9)$$

Il reste à déterminer γ . or, en régime permanent, $\underline{u} = 0$, et $\underline{X}(\infty)$ est donné par l'équation (III-2) dans laquelle on fait $\dot{\underline{X}} = 0$. On a ainsi, avec l'équation (III-9), deux équations fournissant γ (ce qui peut être obtenu directement lors de la vérification statique en simulation sur calculateur analogique.

III - 2 - 3 Essai en simulation de la commande optimale $\underline{u}$

Ayant obtenu la commande optimale $\underline{u}$ à partir de la théorie de KALMAN, nous allons essayer le système (III-1) en simulation, afin d'une part de tester la méthode optimale, en présence de perturbations (échelons et bruit), et d'autre part de dégager des enseignements permettant ultérieurement de quantifier état et commande.

A - Calcul des équations "machine"

Les calibres utilisés sont les suivants :

p_s cal = 5 bars	$\dot{p}_s$ cal = 0.02 b/s
t_s cal = 100°C	$\dot{t}_s$ cal = 100°C/s
q_s cal = 200 kg/h	$\dot{z}_s$ cal = 200°C/s
q_c cal = 200 kg/h	$\dot{q}_c$ cal = 10 kg/h/s
z^d cal = 200 ($\frac{1}{s}$)	$\dot{q}_d$ cal = 10 kg/h/s
δq_s cal = 2000 kg/h/s	
q_s cal = 2000 kg/h	

D'où les équations machines calibrées :

$$\left(\frac{t_s}{100}\right) = \frac{1}{150} - \left(\frac{t_s}{100}\right) + 1.6\left(\frac{q_c}{200}\right) - 2\left(\frac{q_s}{2000}\right) - \frac{1}{2}\left(\frac{z}{200}\right) + 0.36 \quad dt$$

$$\left(\frac{\dot{z}}{200}\right) = \frac{1}{50} \int \left[-\left(\frac{z}{200}\right) + 0,8 \left(\frac{q_d}{200}\right) - 0,16 \right] dt$$

$$\left(\frac{\dot{p}_B}{0,02}\right) = 3,4 \left(\frac{q_c}{200}\right) - 2,81 \left(\frac{q_s}{2000}\right) + 0,281 \left(\frac{q_d}{200}\right)$$

$$\left(\frac{p_B}{5}\right) = \frac{0,02}{5} \int \left(\frac{\dot{p}_B}{0,02}\right) dt, \quad \left(\frac{p_s}{5}\right) = \left(\frac{p_B}{5}\right) - 0,856 \left(\frac{q_s}{2000}\right)^2$$

Et pour les commandes :

$$\begin{aligned} \left(\frac{u_1}{10}\right) = & -0,20620 \left(\frac{p_s}{5}\right) - 0,76953 \left(\frac{q_c}{200}\right) - 1,17636 \left(\frac{t_s}{100}\right) + 0,45077 \left(\frac{z}{200}\right) + 0,14797 \left(\frac{q_d}{200}\right) \\ & + 0,32155 + 1,03634 \left(\frac{\delta q_s}{2000}\right) \end{aligned}$$

$$\begin{aligned} \left(\frac{u_2}{10}\right) = & -0,30128 \left(\frac{p_s}{5}\right) + 0,14797 \left(\frac{q_c}{200}\right) + 0,58866 \left(\frac{t_s}{100}\right) - 0,35647 \left(\frac{z}{200}\right) - 0,46881 \left(\frac{q_d}{200}\right) \\ & + 0,09720 - 0,16414 \left(\frac{\delta q_s}{2000}\right) \end{aligned}$$

Les commandes calibrées et le vecteur ϕ linéarisé ont été calculés sur le calculateur numérique, à l'aide d'un programme FORTRAN donné en annexe.

Enfin,

$$\left(\frac{q_s}{200}\right) = 0,05 \int \left(\frac{u_1}{10}\right) dt$$

$$\left(\frac{q_d}{200}\right) = 0,05 \int \left(\frac{u_2}{10}\right) dt$$

Signalons que le problème a été accéléré par 5, ce qui revient à multiplier par 5 le gain des intégrateurs.

B - Câblage

Le câblage de ces équations est donné figure 26a et 26 b

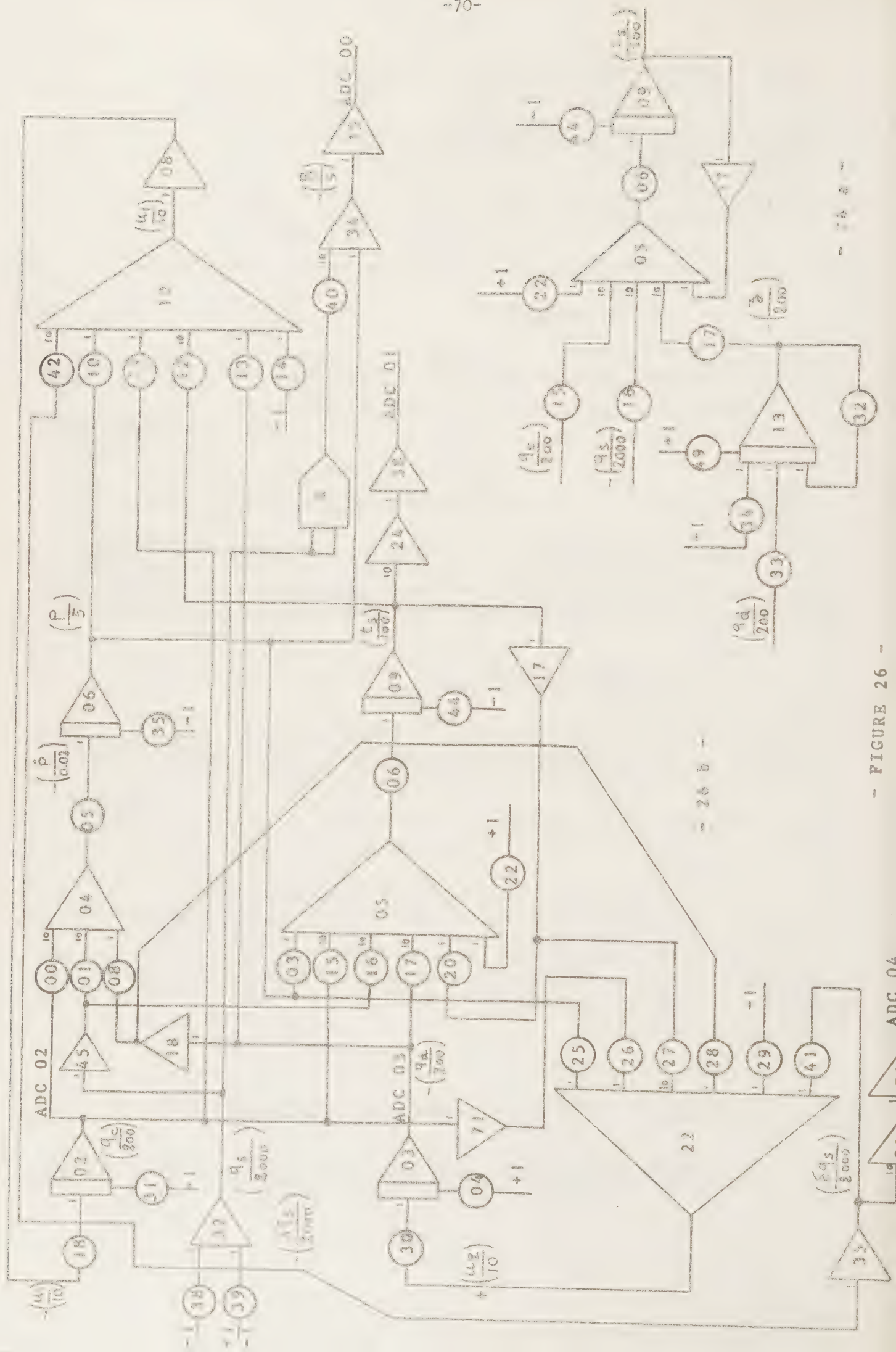
La figure 26a montre le câblage correspondant à la présence explicite de z dans les équations.

La variable d'état z étant la seule non accessible, il nous a paru intéressant d'étudier les résultats d'une optimisation dans laquelle on fait l'approximation $z = 0,8 (q_d - 40)$ ce qui permet d'effectuer un bouclage sur q_d au lieu de z (figure 26 b).

Soit :

$$\left(\frac{u_1}{10}\right) = -0,20620 \left(\frac{p_s}{5}\right) - 0,76953 \left(\frac{q_c}{200}\right) - 1,17636 \left(\frac{t_s}{100}\right) + 0,5080 \left(\frac{q_d}{200}\right) + 0,25160 + 1,03634 \left(\frac{\delta q_s}{2000}\right)$$

$$\left(\frac{u_2}{10}\right) = -0,30128 \left(\frac{p_s}{5}\right) + 0,14797 \left(\frac{q_c}{200}\right) + 0,58866 \left(\frac{t_s}{100}\right) - 0,74881 \left(\frac{q_d}{200}\right) + 0,15160 - 0,16414 \left(\frac{\delta q_s}{2000}\right)$$



- FIGURE 26 -

C - Résultats

La figure 27 donne l'enregistrement des réponses en température et pression à une perturbation de débit de vapeur (Echelon positif de 200kg/h autour de $q_{sN} = 1000$ kg/h).

La figure 27_a correspond à la présence explicite de z dans les équations des commandes.

La figure 27_b correspond à un bouclage non effectué sur z .

La comparaison des deux résultats montre la multitude des réponses du point de vue des amplitudes aussi bien en pression qu'en température.

Du point de vue dynamique, les temps de réponse sont légèrement plus importants dans 27_b, mais l'imprécision des mesures justifie pleinement la non observation de z , ce qui nous permet d'élaborer une commande quasi optimale en boucle fermée sur des grandeurs toutes mesurables.

III - 2 - 4 Apprentissage

On désire établir une commande sous-optimale à trois niveaux (-1,0,+1) à partir de la connaissance d'un certain nombre de solutions optimales exactes obtenus précédemment.

Le vecteur de mesure étant de dimension 5 ($p_s, t_s, q_c, q_d, \delta q_s$), il n'est pas possible de visualiser les trajectoires optimales et l'on est obligé de déterminer les surfaces de commutation de la manière suivante :

Les enregistrements, en simulation, sur une période de quelques heures de fonctionnement réaliste et perturbé, des commandes optimales u_1 et u_2 donnent l'amplitude de ces commandes, à partir desquelles on fixe un seuil de quantification. Ayant quantifié les domaines de variation plausible de chaque variable de mesure, on choisit une valeur dans chaque zone. A chaque échantillon de cinq valeurs correspond une commande optimale $\bar{u}(p_s, t_s, q_c, q_d, \delta q_s)$ qui est quantifiée selon le critère précédemment fixé :

$$\hat{u} = +1 \text{ si } u \geq \frac{u}{2} \text{ (valeur moyenne)}$$

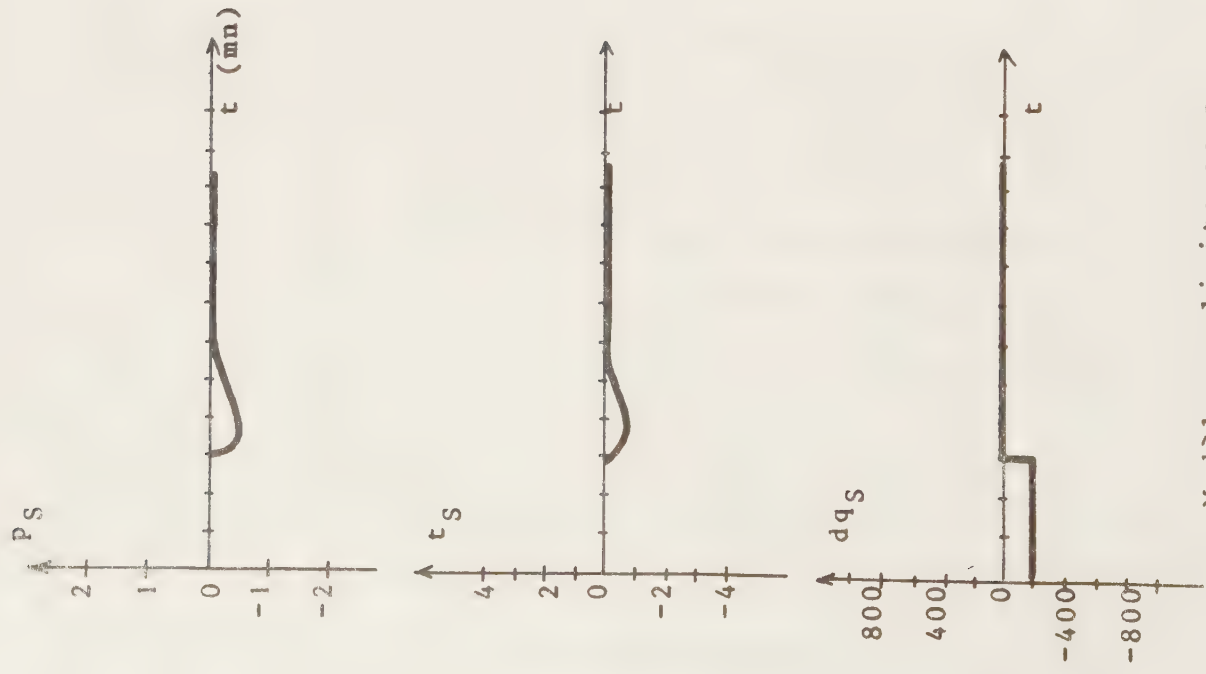
Les trois valeurs de quantification déterminent implicitement dans l'espace à 5 dimensions, pour chaque commande, deux hypersurfaces de séparation. L'apprentissage consiste à adapter les vecteurs de pondération correspondant à chacune des deux configurations associées à u_1 et u_2 de manière à classer correctement l'ensemble des échantillons choisis.

A - Quantification - choix des échantillons

Quantification des états

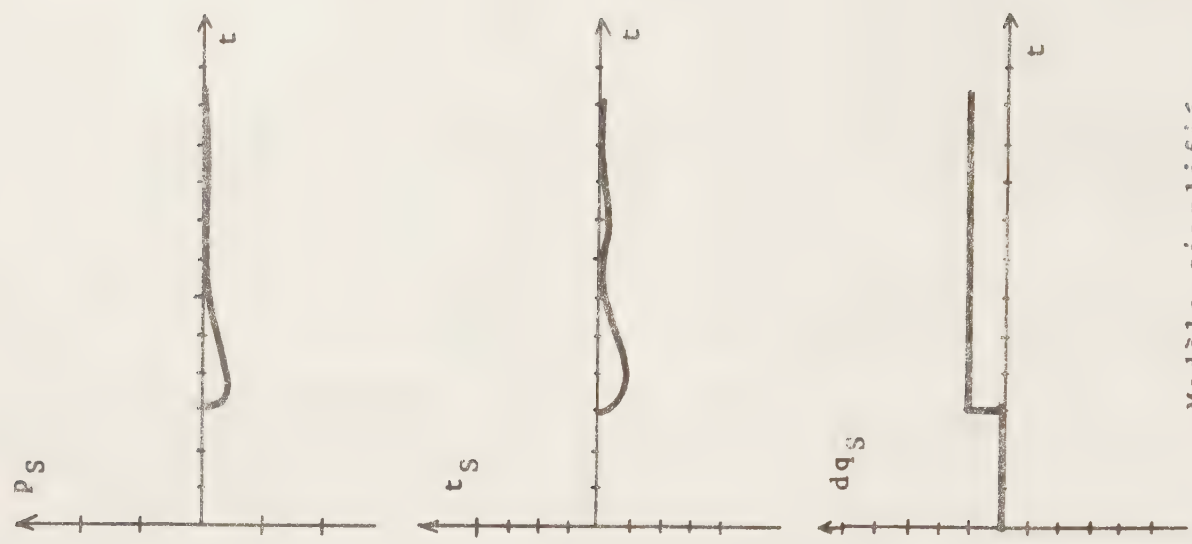
Le choix des zones de quantification des 5 variables d'état a été fait de la manière suivante :

- En tenant compte des écarts de régulation admissibles en t_s et p_s
- Des valeurs de régime permanent de q_c et q_d correspondant à des variations de charge de 20% autour de $q_{sN} = 100$ kg/h (ainsi pour $q_s = 800, 1000$ et 1200 kg/h, q_c vaut approximativement 60,78 et 96 kg/h)
- De l'influence relative des différents débits sur les variables réglées p_s et t_s
- Des essais préliminaires menés sur chaudière en télé-réglage, desti-



Modèle explicite en z
- 27 a -

Réponses à un échelon
de 200 kg/h de vapeur

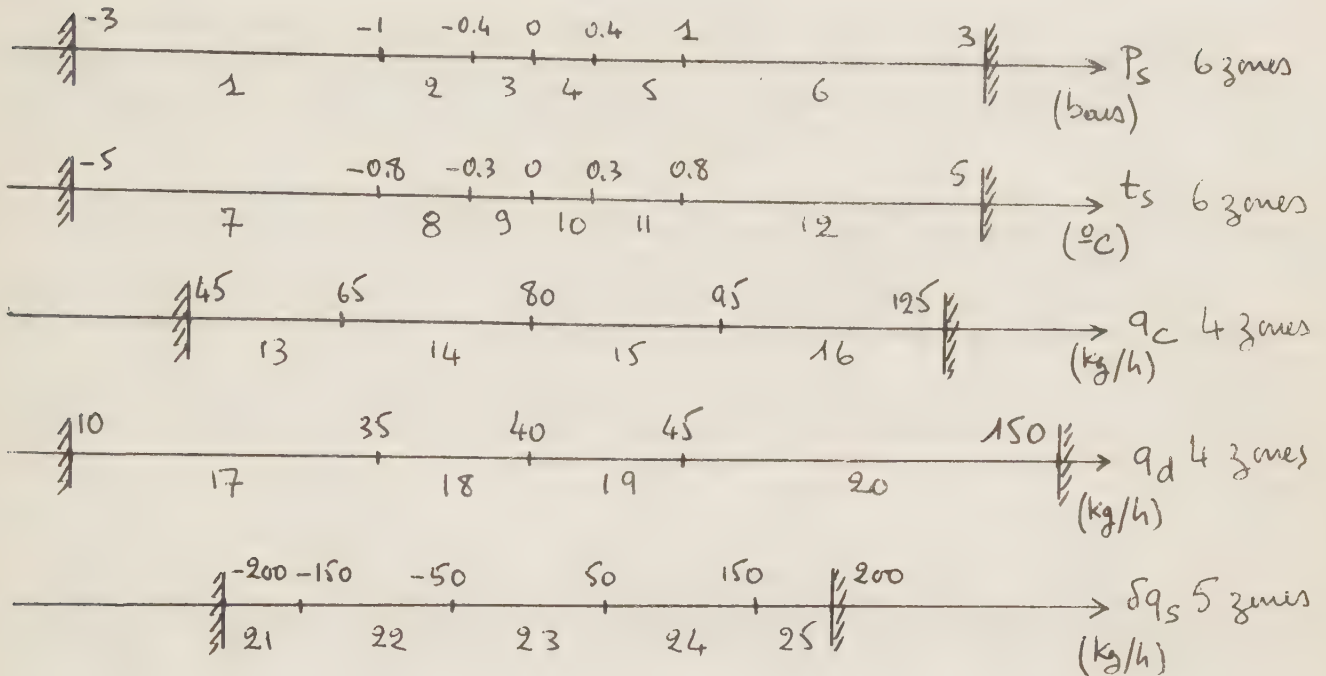


Modèle simplifié
- 27 b -

Simulation de la commande
optimale obtenue à partir
de la théorie de KALMAN

nés à définir les points de fonctionnement les plus fréquents.

Ces critères de choix ont donné les résultats suivants :

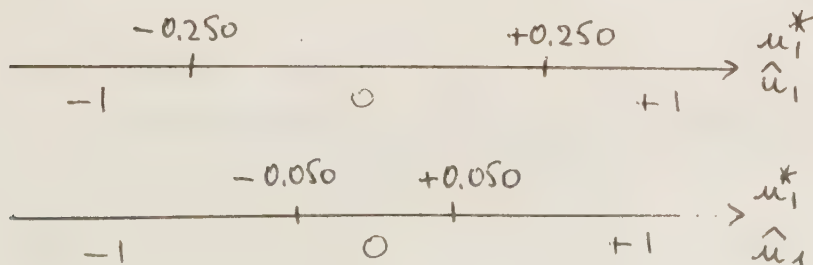


Cette quantification divise l'espace d'état en $6 \times 6 \times 4 \times 4 \times 5 = 2880$ hypercubes.

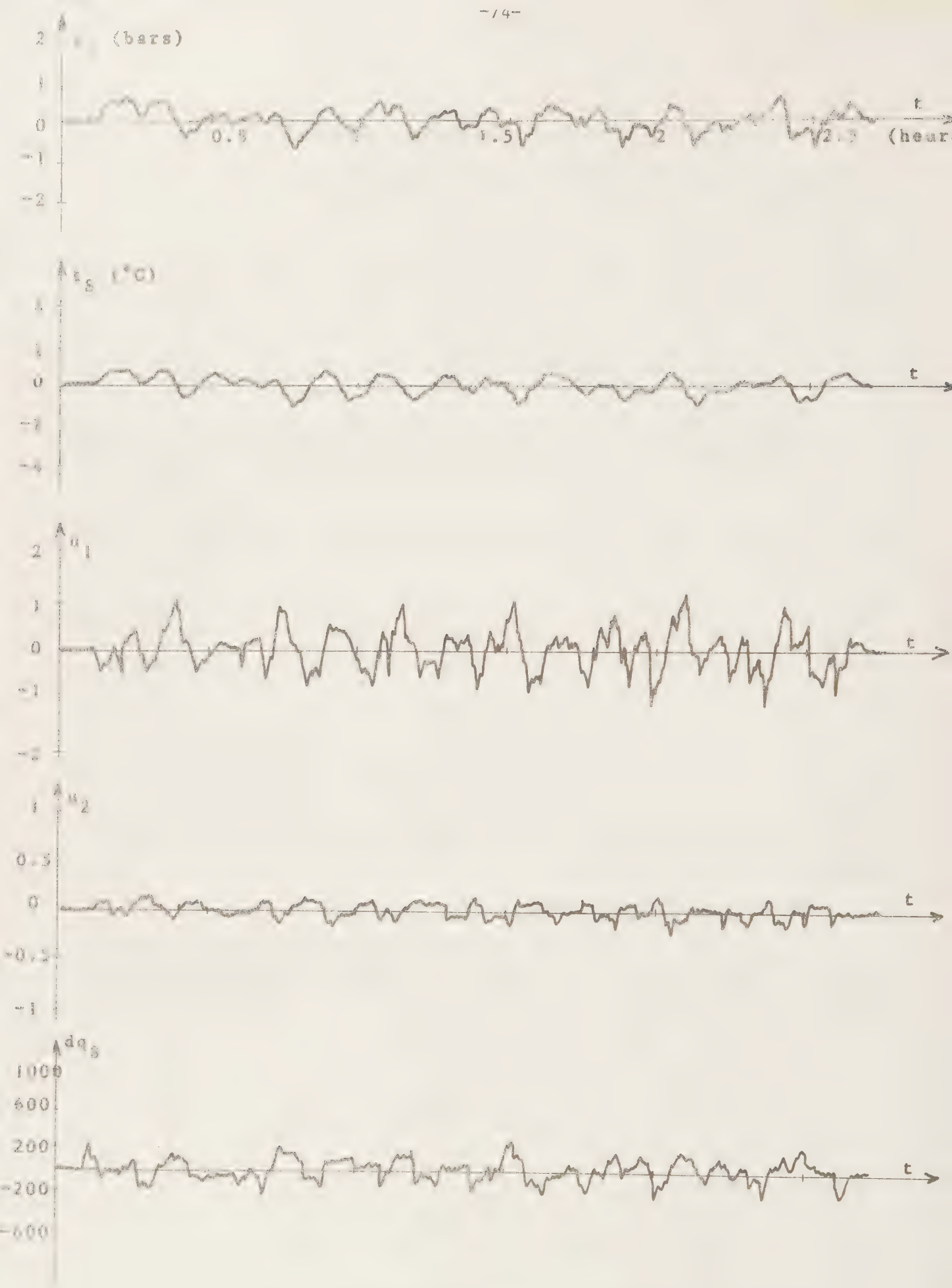
Quantification des commandes

Comme nous l'avons indiqué plus haut, le choix des zones de quantification des commandes a été fait à partir de l'étude des enregistrements des commandes optimales fournies par la théorie de KALMAN, en présence de perturbations aléatoires sur le débit de vapeur. (Fig. 28)

Les résultats sont donnés en (kg/h)/s :



En fait le choix de ces quantifications est difficile, car il a une



Perturbations aléatoires sur dq_S
 Réponses optimales en u_1 et u_2

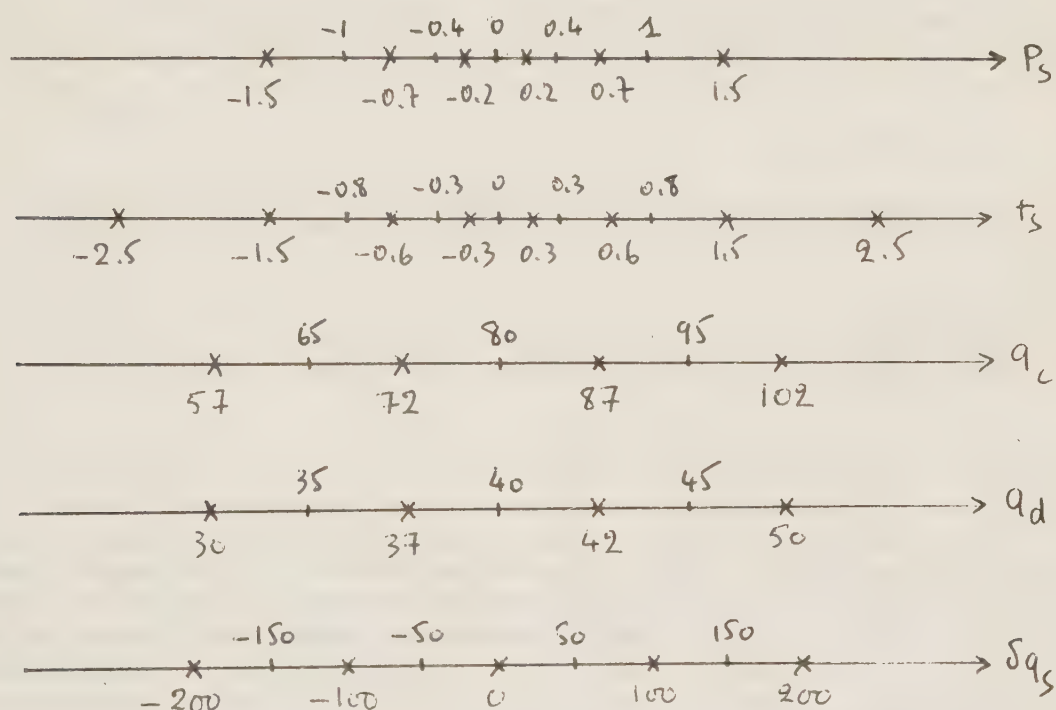
grande importance dans la mesure où il influence le positionnement des hyper-surfaces de séparation. Nous avons pu remarquer en particulier au cours de nos essais, que le fait de changer ± 0.250 en ± 0.260 pour la quantification de u_1 changeait d'une manière non négligeable de déroulement de l'Apprentissage (donc les valeurs des vecteurs de pondération) et par conséquent modifiait les commandes sous-optimales résultantes.

En définitive, on peut fixer à priori au vu d'enregistrements, un ordre de grandeur des quantifications, mais il est nécessaire, dans l'état actuel des choses d'effectuer plusieurs essais permettant de déterminer la meilleure solution ou du moins la plus satisfaisante.

Choix des échantillons

Dans un premier temps, les essais en simulation et sur processus réel n'ayant pour but que de tester la méthode d'apprentissage, nous n'avons pas cherché à ne considérer qu'un nombre minimum de solutions optimales exactes.

Nous avons pris l'ensemble des échantillons (un par hypercube) augmenté de deux échantillons supplémentaires en t_s , comme indiqué sur le schéma ci-dessous.



Soit au total 3840 échantillons.

B - Le calcul hors ligne algorithme

Le but à atteindre peut être défini de la manière suivante :

- u_1 et u_2 doivent être fournis en ligne par un générateur de fonction adaptatif sous la forme d'un signal à trois niveaux (-1, 0 et +1)

- Le calcul des vecteurs de pondération correspondant sera effectué hors ligne, et le générateur de fonction sera simulé sur calculateur numérique.

- Le facteur de qualité p_{cc} devra excéder 98% ($P_{cc} \geq 0,980$)

Cela signifie que, n'ayant à priori aucune indication sur la forme des hypersurfaces de séparation, nous ne pouvons appliquer aucun des critères de convergence, qui assurent un classement correct de l'ensemble des échantillons présentés. Nous choisissons alors une valeur minimum du facteur de qualité, valeur à partir de laquelle, si on le désire, on peut arrêter l'apprentissage. En effet, une erreur de classement de 2% n'a que très peu d'influence sur une commande en ligne, principalement en présence de perturbations.

- La commande à trois positions en u_1 et u_2 est réalisable à l'aide de deux générateurs de fonction pour chaque commande. Nous devons donc simuler le fonctionnement de 4 générateurs adaptatifs et trouver 4 solutions en w .

Pour u_1 : w_{A1} et w_{B1}

Pour u_2 : w_{A2} et w_{B2}

Nous avons choisi l'algorithme à relaxation pour calculer ces vecteurs. Le calcul est mené simultanément pour les 4 vecteurs, selon le programme donné en annexe.

C - Résultats

Le calcul a été mené sur I B M 360 - 25 (durée 50 minutes). Après plus de 1 000 corrections, les valeurs de P_{cc} sont les suivantes :

Pour w_{A1}	: p_{cc}	= 0.980
w_{B1}	: p_{cc}	= 0.982
w_{A2}	: p_{cc}	= 0.989
w_{B2}	: p_{cc}	= 0.987

Une remarque s'impose : Nous avons fait un essai d'adaptation jusqu'à plus de 25 000 corrections sans améliorer sensiblement le facteur de qualité du classement. On peut en conclure que les hypersurfaces de séparation ne satisfont pas certains critères de convergence. Cependant, comme nous l'avons dit précédemment, l'erreur de classement commise a peu d'influence sur la commande sous-optimale résultante. (Rappelons que cette erreur de classement est au maximum de 2% et au minimum de 1,1%).

III - 2 - 5 Essai en simulation de la commande sous-optimale $\hat{u}$

A - But de cet essai

Après avoir calculé par la méthode de reconnaissance de forme les vecteurs de pondération des générateurs de fonction adaptatif, nous testons la commande sous-optimale résultante $\hat{u}$ sur le processus simulé, décrit par les équations d'état données plus haut. Ce test a pour but d'évaluer en simulation la commande sous-optimale appliquée d'une part au modèle contenant explicitement

z (simulation 2), d'autre part au modèle où z est remplacé par son expression fonction de q_d ($z = 0,8 (q_d - 40)$ (simulation 1)), dans l'équation de température du modèle.

Cette simplification se justifie par le fait qu'en réalité la constante de temps en z existe plus ou moins dans l'asservissement de commande. On peut donc ne pas s'inquiéter outre mesure des résultats obtenus en simulation (cf. paragraphe C suivant), les résultats obtenus sur le processus réel confirmant le modèle adopté.

B - Câblage

Les essais en simulation sont menés de la manière suivante :

Les générateurs de fonction adaptatifs sont simulés sur ordinateur hybride EAI 640 (quantification des entrées, pondération, sorties, partielles, fonction logique, sortie binaire). Les vecteurs de pondération sont introduit en mémoire à l'appel d'un programme principal (donné en annexe).

Les équations d'état du processus simulé sont câblées sur calculateur analogique TR 48. Les mesures des grandeurs analogiques (t_s , p_s , q_s , q_d , δq_s) sont transformées en signaux binaires dans des blocs A/D et envoyés en continu sur l'ordinateur qui effectue en permanence les calculs X_w . Les commandes binaires (-1, 0, +1) qui en résultent sont converties en signaux analogiques dans des blocs D/A et complètent le bouclage après réglage de la pondération des niveaux +1 et -1 pour chaque commande $\hat{u}_1$ et $\hat{u}_2$.

Le schéma de ce câblage est donné figure 29

C Résultats

La figure 30a montre les réponses en p_s et t_s à un échelon positif de vapeur ($\delta q_s = 200$ kg/h), dans le cas de simulation N°1, tandis que la figure 30b représente le résultat de la simulation N°2 pour un échelon identique.

Dans les deux cas, on remarque que le régime permanent en p_s et t_s s'établit sur une plage autour de zéro. Ceci est dû au principe même de la méthode sous-optimale, les grandeurs étant quantifiées. La largeur de cette plage peut être rendue plus faible en réduisant le pas de quantification de p_s et de t_s autour de zéro. Cependant, la largeur minimum sera égale au niveau de sensibilité des capteurs ($\pm 0,3$ bars pour le capteur de pression, quelques 1/10 de degré pour les thermo-couples) et le choix des quantifications à $\pm 0,4$ bars et $\pm 0,3^\circ\text{C}$ tient compte de ce fait.

Une deuxième remarque, est que les réponses à un même échelon sont relativement reproductibles en pression dans les deux cas de simulation, alors qu'il n'en est pas de même pour la température. En effet, l'équation en z introduit des transitoires (pompage) que l'on supprime en remplaçant z par son expression en q_d dans l'équation de température. On peut signaler enfin l'influence de la pondération des commandes $\hat{u}_1$ et $\hat{u}_2$ sur la dynamique de p_s et t_s . Après essais préliminaires, les valeurs de pondération de $\hat{u}_1$ et $\hat{u}_2$ ont été fixées pour cette simulation à 0,200 et 0,100.

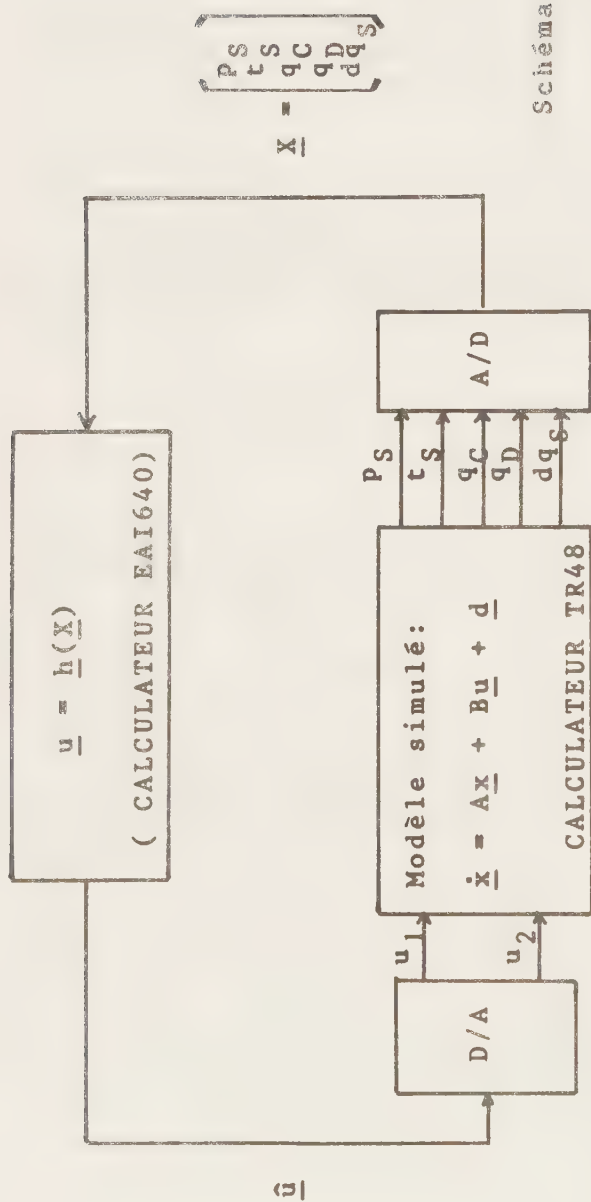
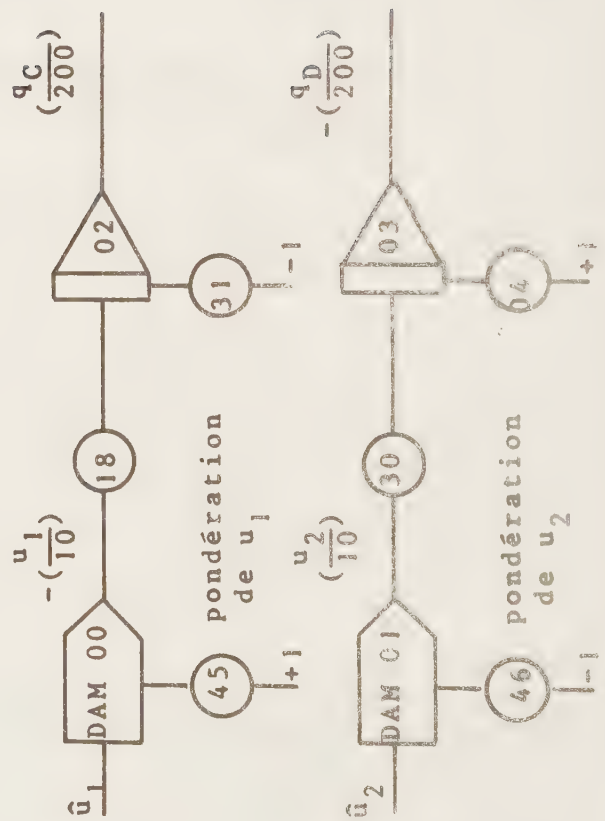
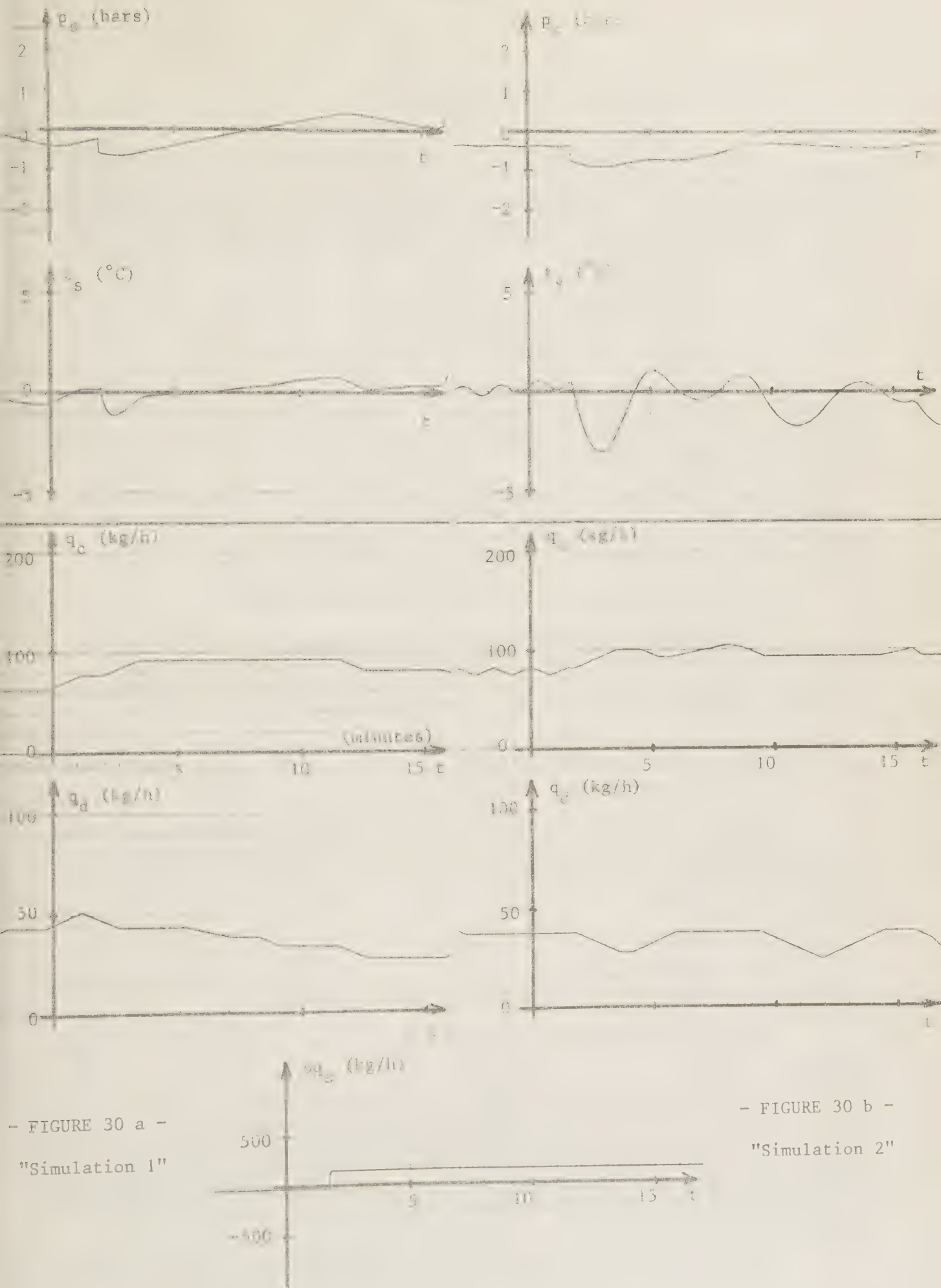


Schéma de câblage général



Câblage des commandes
sous-optimales (TR 48)

- FIGURE 29 -



- FIGURE 30 a -
"Simulation 1"

- FIGURE 30 b -
"Simulation 2"

III - 2 - 6 Application au processus pilote

Les essais en simulation nous ont permis de tester le modèle mathématique mis au point au chapitre I. Nous avons vu que ce modèle pouvait être simplifié, en abaissant le degré de l'équation de température par la suppression de la dérivée $\dot{z}$. De plus, nous avons pu ajuster la pondération des commandes quantifiées $\bar{u}_1$ et $\bar{u}_2$. Nous allons maintenant tester ces commandes en ligne sur le processus réel.

A - Mise en oeuvre

Cet essai est réalisé selon le schéma de la figure 31.

Les générateurs de fonction adaptatif, sont simulés sur ordinateurs hybride EAI 640. Cette simulation comprend : la quantification des états (entrées) la pondération, les sommations partielles, la génération des commandes intermédiaires u_{Ai} et u_{Bi} , la logique $u_i = \text{sgn} (u_{Ai} + u_{Bi})$.

Le calculateur analogique TR 48 sert à la fois à l'acquisition des données (traitement des mesures p_s et t_s , q_s , q_1 et δq_1), de convertisseur A/D pour ces données, et de convertisseur D/A pour les commandes $\bar{u}_1$ et $\bar{u}_2$ ainsi qu'à pondérer les commandes analogiques résultantes.

La chaudière est tout d'abord pilotée manuellement.

Le programme de calcul numérique des commandes sous-optimales étant mis en route, ces commandes ne sont appliquées à la chaudière que lorsque la pression et sa température de la vapeur surchauffée évoluent dans un domaine où elles sont commandables (autour de 20 bars et 300°C).

La charge étant réglée à sa valeur nominale $q_{sN} = 1000$ kg/h, on commute alors en commande automatique et on laisse le système évoluer jusqu'à son régime permanent en p_s et t_s .

Lorsque le système est stabilisé, on étudie les réponses en pression et en température à des perturbations de charge. Ces perturbations sont de deux types : perturbations en échelon (20% de q_{sN}) et perturbations "aléatoires" simulant un téléériglage.

B - Résultats

La figure 32 montre une réponse en p_s et t_s à deux échelons de vapeur de 200kg/h autour de 1000 kg/h, (échelons positif et négatif).

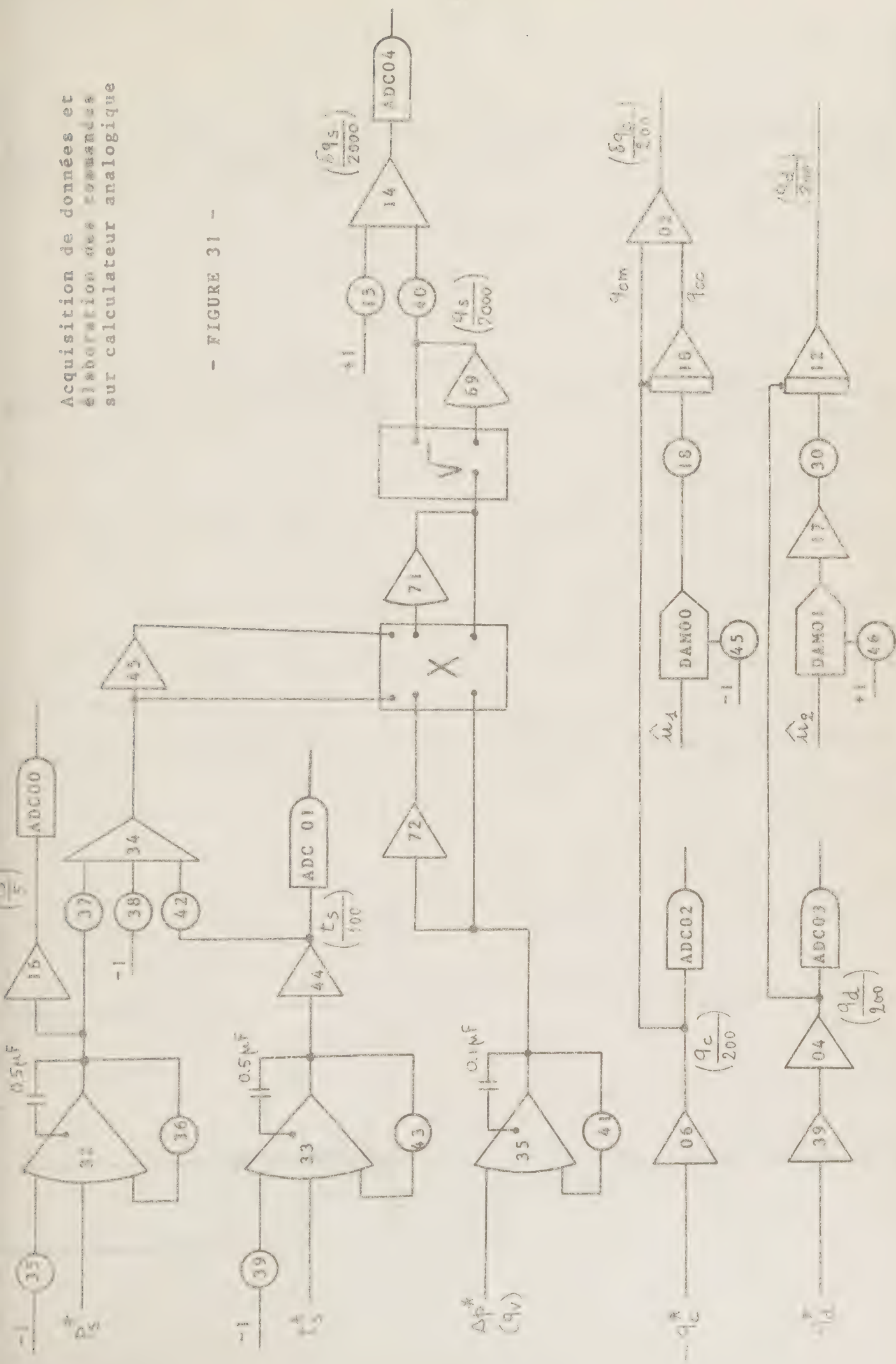
La figure 33 montre le comportement de p_s et t_s en présence de bruit sur q_s .

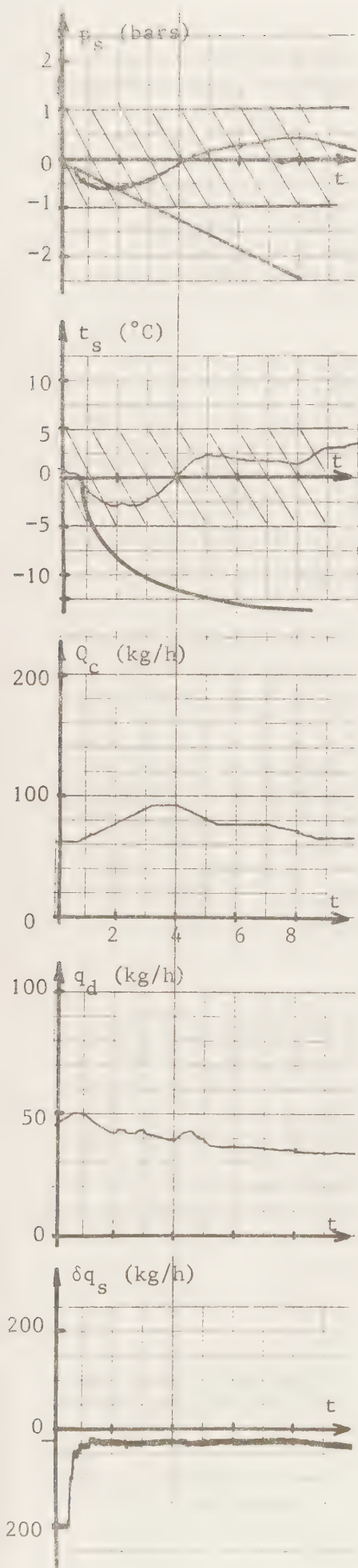
Si l'on compare ces résultats avec les résultats de la simulation données précédemment, on remarque que les temps de réponse en pression et température, pour des échelons identiques de vapeur, sont légèrement plus importants en commande réelle (cela étant dû à une approximation des constantes de temps du modèle). En ce qui concerne les amplitudes, les réponses en pression sont du même ordre de grandeur, tandis que les réponses en température ont une amplitude plus grande en commande réelle.

L'enregistrement effectué en présence de bruit sur la vapeur montre

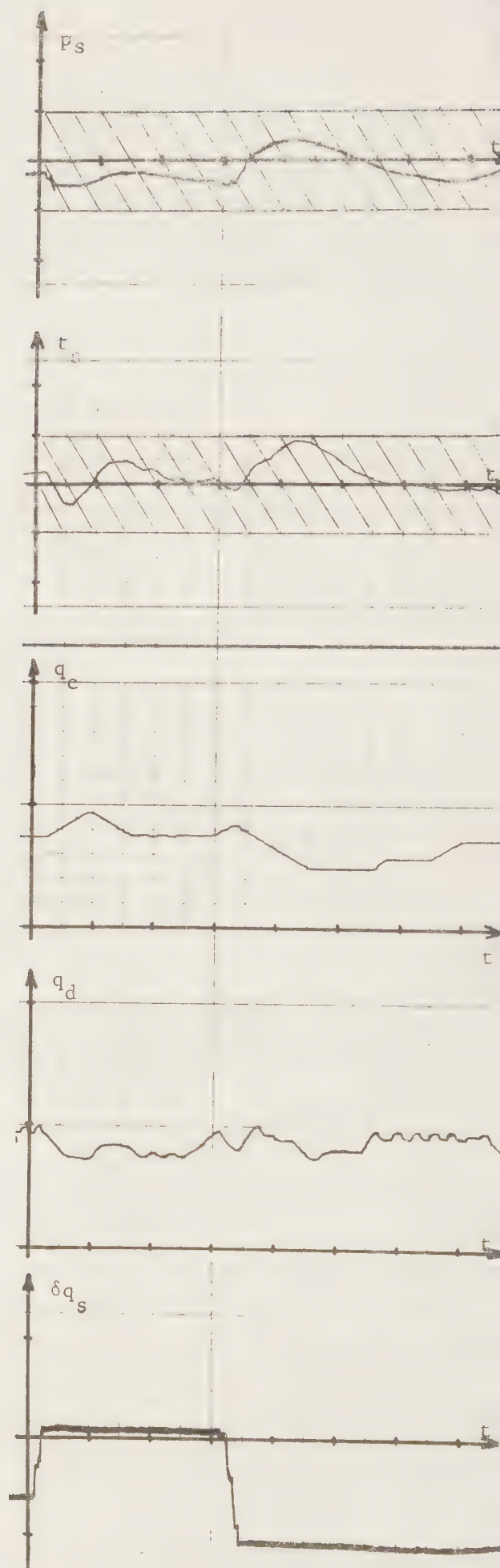
Acquisition de données et
élaboration des commandes
sur ordinateur analogique

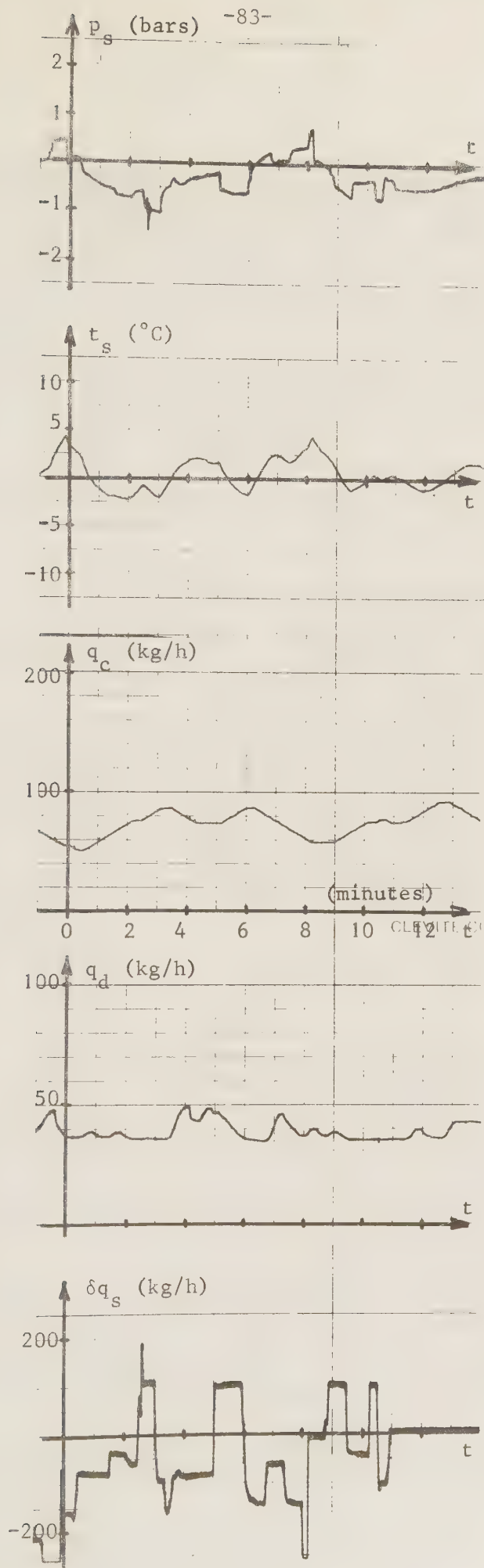
- FIGURE 31 -





- FIGURE 32 -





- FIGURE 33 -

que le système évolue de telle sorte que les écarts maximum admissibles (± 1 bar $\pm 5^{\circ}\text{C}$, donnés au paragraphe(I-1-5) sont ici respectés ($\pm 0,8$ bars , $\pm 3^{\circ}\text{C}$). Ces résultats apparaissent donc tout à fait satisfaisants.

CONCLUSION

A partir de la connaissance de solutions optimales exactes (trajectoires ou commandes), nous avons mis en oeuvre une commande sous-optimale en boucle fermée dont la simplicité de structure est sans doute la principale caractéristique. En effet, une fois la quantification des entrées (état) réalisée, le programme FORTRAN de calcul en ligne du vecteur de commande n'exécute pas 10 instructions. Le temps de calcul est négligeable (quelques dixièmes de seconde pour la totalité du programme) et l'ordinateur peut fonctionner sur interruptions (20 secondes par exemple pour notre chaudière, dont le temps de réponse en boucle fermée est de 200 à 250 secondes).

Cette méthode a également été testée sur un problème simple comportant des contraintes et les résultats obtenus sont, comme nous l'avons vu, très encourageants. (la structure n'est pas compliquée par l'adjonction des butées).

De plus, des essais que nous n'avons pas mentionné dans cette étude, nous ont permis, d'une part de réduire considérablement le nombre d'échantillons utilisés dans la phase d'apprentissage (problème de la généralisation), d'autre part de réduire la dimension des vecteurs de pondération.

La variété des moyens de calcul mis à notre disposition (IBM 1130, 360-25, 360 - 40, EAI 640), et leur capacité de mémoire, ont rendu cette étude possible et la mise en oeuvre des différentes phases plus aisée. Cependant, il eut été parfaitement possible de mener à bien la mise au point de la commande sous-optimale tant en ce qui concerne la phase d'apprentissage hors ligne, que la phase de commande en ligne, au moyen de régulateurs autonomes, analogiques et logiques, de technologie très simple (codeurs et comparateurs pour la quantification, potentiomètres asservis pour la pondération, additionneurs et relais sensibles pour la génération des sorties). La qualité et la fiabilité des composants existant actuellement sur le marché Français, ainsi que leur prix de revient permettent d'envisager la construction d'un prototype comme tout à fait réalisable.

Il nous semble également possible d'aller plus avant dans la notion d'apprentissage : dans la mesure où un critère d'optimisation ne représente souvent qu'un "critère de bon fonctionnement", les informations nécessaires pour effectuer l'apprentissage peuvent à la limite provenir de l'expérience acquise par les ingénieurs et les techniciens travaillant sur le processus et non d'algorithmes complexes de calcul. Ces deux points de vue nous ont d'ailleurs guidé au cours de notre étude : en effet, le choix des solutions optimales exactes, n'a pu être fait, en l'absence de tout support géométrique tel que la représentation des trajectoires dans l'espace d'état, qu'à partir de la connaissance intuitive des réactions prévisibles du processus pilote, connaissance due à l'expérience acquise de longue date, l'établissement du modèle mathématique étant lui-même lié à cette connaissance. Quant au critère d'optimisation, il est bien évident que sa complexité ne peut qu'aller croissant pour des unités industrielles importantes. Son rôle devra donc être nécessairement limité.

La méthode de reconnaissance de formes, liée à la mise en oeuvre de systèmes à apprentissage, paraît particulièrement bien adaptée à la commande de

processus soumis à des perturbations aléatoires (télé-réglage). La discrétisation des données, mise en jeu par cette méthode, en effet, rend l'obtention d'un régime calme impossible dans la plupart des cas. PAR contre, cette quantification permet de se satisfaire de mesures approchées et donc de capteurs simples et peu coûteux. A la limite, ces capteurs peuvent être de conception très grossière (par exemple systèmes photo-électriques pour la mesure de niveaux). Enfin, la quantification des commandes, réduites ici à leur plus simple expression (-1, 0, +1) mais que l'on peut édulcorer, à comme conséquences un accroissement de la durée de vie des organes de commande (servo-moteurs, vannes électro-pneumatiques etc...) et un abaissement de leur coût en énergie.

Ces différentes raisons font de cette méthode un outil bien adapté aux problèmes que peut poser l'automatique industrielle.

Nous rappelons ci-dessous les phases principales de la méthode :

1 - Définition d'un modèle et d'un critère, afin d'utiliser une méthode d'optimisation pour avoir un échantillon optimal exact (nous avons utilisé ici la théorie de KALMAN et un modèle d'Etat linéarisé, mais ceci n'est pas nécessaire et en particulier nous aurions pu tout aussi bien utiliser la programmation non linéaire...) . Cette première phase permet d'obtenir certaines informations primaires.

2 - Quantification des grandeurs mesurées et des commandes en fonction de la connaissance que l'on a du processus, des résultats de l'optimisation (première phase) et des buts que l'on désire atteindre (coûts, prépondérance de certaines variables réglées etc...).

3 - Apprentissage (certains diraient "régression") hors ligne, c'est-à-dire obtention, à l'aide d'algorithmes de reconnaissance de formes, des coefficients de la commande sous-optimale.

4 - Application et mise au point de cette commande sous-optimale en simulation (cette mise au point peut être effectuée à partir de considérations intuitives ou par expérience acquise).

5 - Mise en ligne de la commande sous-optimale avec le processus réel,

BIBLIOGRAPHIE

CHAPITRE I

- (1) J. RICARD Equipement thermique des usines génératrices d'énergie électrique . DUNOD 1961
- (2) R. CARDINAL et J. GLEVAREC Marche industrielle a faible excès d'air de la chaudière tranche 4 de la Centrale thermique de NANTES-CHEVRE.
- (3) C. LAURGEAU Etude d'un réacteur chimique pilote : thèse octobre 1969 à NANTES (Page 18 - 20)
- (4) J. F. LE CORRE Régulation d'une chaudière : thèse janvier 1969 NANTES.
- (5) E.D.F. Service d'Essais spéciaux , rapports provisoires 1 à 5. 1966
- (6) M. AZEMA Introduction à l'étude de la surchauffe :Rapport D.E.A. octobre 1969 NANTES
- (7) Ph. DE LARMINAT, Y. THOMAS Rapport D.G.R.S.T. juin 1970 NANTES.

CHAPITRE II

- (1) J.C. MARC "Comman~~d~~de optimale avec butée " D.E.A. juin 1969 NANTES
- (2) J.M. MENDEL, J.J. ZAPALAC "Self organizing Control Systems Vol. 3 " DOUGLAS REPORT SM 51975 Février 1966
Missile & Space Systems divisions D.A.C. California
- (3) B. WIDROW " Pattern Recognition and Adaptive Control " I.E.E.E. Transactions on Automation and Industry Vol. 83 pp. 269-277 Septembre 1964
- (4) F.W. SMITH "Control by adaptive pattern-recognition techniques" Technical Report SU-SEL 64002 (TR 6762 - 1)
STANFORD ELECTRONICS LABORATOIRES California April 1964
- (5) F.W. SMITH "A trainable non linear fonction generator" I.E.E.E. transactions on Automatic Control Vol. AC-11 N°2 April 1966.

- (6) J.W. BERKOVEC D.L. EPLEY " On time optimal control with threshold logic units " 1964 WESCON, August 1964.
- (7) J.J. ZAPALAC " Self organizing control systems " Vol. 1
Douglas Report SM 47857 July 1965
Missile and space Systems Division
DOUGLAS AIRCRAFT CORP. Santa Monica . California
- (8) C.H. MAYS " Adaptative threshold logic "
Technical report SEL 63-027 (TR 1557-1) April 1963
STANFORD ELECTRONICS LABORATORIES California
- et: " Effects of adaptation parameters on convergence time and tolerance for adaptative-threshold elements "
IEEE TRANS. on ELECT. COMP. Short Notes pp.465-478
August 1964

CHAPITRE III

- (1) Ph. de Larminat , Y. Thomas " Conditions d'optimalité pour les problèmes de butées et de contraintes "
Rapport DGRST Annexe 4 - NANTES - Juin 1970
- (2) A. ABDALLAH Rapport provisoire - Laboratoire d'Automatique - Juin 1970
- (3) R.E. KALMAN " Contributions to the theory of Optimal Control "
Bolletin Societat Matematica Mexicana pp 102-119 (1960)
- (4) BEYSSENS , DESCUSSE Rapport de projet, Laboratoire d'Automatique
ENSM - NANTES - Mai 1970
- (5) Y. THOMAS, Ph. de LARMINAT " Application du principe de séparation à la commande d'un Générateur de vapeur "
Rapport DGRST Annexe 5 - NANTES - Juin 1970

A N N E X E

Programme FORTRAN utilisés dans cette étude :

- A. Programme de calcul de K et ϕ (théorie de KALMAN)*
- B. Programme de calcul des équations calibrées donnant u_1 et u_2*
- C. Programme d'apprentissage (problème à 5 états)*
- D. Programmes d'élaboration d'une bande vecteur - poids WA WB*
 - D - 1 Problème en $1/s^2$*
 - D - 2 Problème à 5 états*
- E. Programmes principaux de calcul en ligne des commandes sous optimales*
 - E - 1 Problème en $1/s^2$*
 - E - 2 Problème à 5 états*
 - Quantification 1*
 - Quantification 2*


```

100 F1=OK
PHASE CHNVX,*
DIMENSION YY(35),DERY(35),PRMT(5),AUX(16,35),BETA(35),Y(35),YI(35)
EXTERNAL FCT,OUTP
COMMON NC,JK,ITEST,A(7,7),ALPHA(7,7),P(7,7),G(7),E(7),D(7),QS,NDIM
101 I=1,NDIM
102 READ(1,101)PRMT(I),I=1,4)
103 READ(1,102)NDIM,NC,JK
104 READ(1,103)Y(I),I=1,NDIM
105 READ(1,104)Y(I),DERY(I),I=1,NDIM
DO 20 I=1,NDIM
20 YI(I)=Y(I)
GO TO(1,2,2),JK
106 I=1,201)NC
GO TO 3
107 WRITE(3,202)NC
GO TO(3,3,33),JK
30 READ(1,100)QS
WRITE(3,203)QS
TF=PRMT(2)
PRMT(2)=PRMT(1)
4 IF(TAU-20.*PRMT(3))42,41,41
2 PRMT(3)=0.5*PRMT(3)
GO TO 4
108 DO 5 I=1,NDIM
YY(I)=Y(I)
PRMT(1)=PRMT(2)
PRMT(2)=PRMT(2)+TAU
IF(PRMT(2)-TF)7,7,6
109 PRMT(2)=TF
CALL SORTIE(PRMT,TF,Y)
ITEST=1
IF(PRMT(1)-TF)8,18,18
110 CALL IHFB(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
VALMAX=ABS(Y(1)-YI(1))
DO 10 I=2,NDIM
VALABS=ABS(Y(I)-YI(I))
IF(VALABS-VALMAX)9,10,10
111 VALMAX=VALABS
112 CONTINUE
SIGMA=0.0
DO 11 I=1,NDIM
BETA(I)=VALMAX/ABS(Y(I)-YI(I))
113 SIGMA=SIGMA+BETA(I)
DO 12 I=1,NDIM
DERY(I)=BETA(I)/SIGMA
I=IHLF+1
GO TO(13,4,4,4,14,14,14,14,14,14,14,14,15,17,17),I
114 PRMT(3)=PRMT(3)+PRMT(3)
GO TO 4
115 PRMT(3)=0.25*PRMT(3)
GO TO 4
15 PRMT(3)=0.00048828125*PRMT(3)
DO 16 I=1,NDIM
16 Y(I)=YY(I)

```

```
GO TO 7
17 WRITE(3,300)IHLF
18 STOP
100 FORMAT(E12.5)
101 FORMAT(4E12.5)
102 FORMAT(3(7X,I3))
103 FORMAT(3E12.5)
104 FORMAT(2E12.5)
201 FORMAT(1H1,30(/),30X,'ETUDE DE L''ESTIMATION',//,50X,I3,2X,'VARIABLE(S) D''ETAT')
202 FORMAT(1H1,30(/),30X,'ETUDE DE L''OPTIMALISATION',//,50X,I3,2X,'VARIABLE(S) D''ETAT')
203 FORMAT(//,50X,'QS = ',E12.5)
300 FORMAT(///,10X,'ERREUR      IHLF = ',I3)
END
```

```

SUBROUTINE SORTIE(PRMT,TF,Y)
DIMENSION PRMT(1),Y(1)
COMMON NC,JK,ITEST,A(7,7),ALPHA(7,7),P(7,7),G(7),E(7),D(7),PS,INDI
GO TO(11,12,12),JK
11 WRITE(3,200)PRMT(1)
GO TO 13
12 T=TF-PRMT(1)
WRITE(3,200)T
13 ISUP=0
L=NC
GO TO(1,2,3,4,5,6,7),NC
7 IINF=ISUP+1
L=L-1
ISUP=IINF+L
WRITE(3,307)(Y(I),I=IINF,ISUP)
6 IINF=ISUP+1
L=L-1
ISUP=IINF+L
WRITE(3,306)(Y(I),I=IINF,ISUP)
5 IINF=ISUP+1
L=L-1
ISUP=IINF+L
WRITE(3,305)(Y(I),I=IINF,ISUP)
4 IINF=ISUP+1
L=L-1
ISUP=IINF+L
WRITE(3,304)(Y(I),I=IINF,ISUP)
3 IINF=ISUP+1
L=L-1
ISUP=IINF+L
WRITE(3,303)(Y(I),I=IINF,ISUP)
2 IINF=ISUP+1
L=L-1
ISUP=IINF+L
WRITE(3,302)(Y(I),I=IINF,ISUP)
1 IINF=ISUP+1
L=L-1
ISUP=IINF+L
WRITE(3,301)(Y(I),I=IINF,ISUP)
GO TO(8,8,9),JK
9 IINF=ISUP+1
ISUP=IINF+NC-1
WRITE(3,400)(Y(I),I=IINF,ISUP)
7 WRITE(3,300)PRMT(3)
RETURN
200 FORMAT(/,10X,'T= ',E14.6)
300 FORMAT(20X,'H= ',E14.6)
301 FORMAT(104X,E12.5)
302 FORMAT(84X,2(4X,E12.5))
303 FORMAT(68X,3(4X,E12.5))
304 FORMAT(52X,4(4X,E12.5))
305 FORMAT(36X,5(4X,E12.5))
306 FORMAT(20X,6(4X,E12.5))
307 FORMAT(4X,7(4X,E12.5))
400 FORMAT(/,3X,'PHI',7(4X,E12.5))
END

```

```

SUBROUTINE FCT(X,Y,DERY)
  DIMENSION Y(1),DERY(1),C(7,7),PDERY(7,7),RAY(7,7),X1(7,7),R(7,7),P
1 HI(7),RPHI(7),F(7),CDE(7)
  COMMON NC,JK,ITEST,A(7,7),ALPHA(7,7),P(7,7),G(7),E(7),D(7),QS,NDIM
  KL=0
  DO 50 K=1,NC
  DO 51 J=K,NC
    I=KL+J
    C(K,J)=Y(I)
51 C(J,K)=C(K,J)
  50 KL=KL+NC-K
    GO TO(10,11,11),JK
  10 CALL PROMAT(NC,A,C,RAY)
    CALL PROMAT(NC,ALPHA,C,RI)
    CALL PROMAT(NC,C,RI,R)
    SIGN=1.
    GO TO 12
  11 CALL PROMAT(NC,C,A,RAY)
    CALL PROMAT(NC,C,ALPHA,RI)
    CALL PROMAT(NC,RI,C,R)
    SIGN=-1.
  12 DO 2 I=1,NC
    DO 2 J=1,NC
      2 PDERY(I,J)=RAY(I,J)+RAY(J,I)+(P(I,J)-R(I,J))*SIGN
    KL=0
    DO 60 K=1,NC
    DO 61 J=K,NC
      I=KL+J
61 DERY(I)=PDERY(K,J)
  60 KL=KL+NC-K
    GO TO(1,1,3),JK
  3 DO 4 I=1,NC
    K=NDIM-NC+I
    PHI(I)=Y(K)
    DO 4 J=1,NC
  4 RAY(I,J)=A(J,I)+RI(I,J)
    CALL PROVEC(NC,RAY,PHI,RPHI)
    DO 5 I=1,NC
  5 F(I)=E(I)*QS+D(I)
    CALL PROVEC(NC,C,F,CDE)
    RAMBDA=QS*QS/1000000.0
    RAMBDA=1.225*RAMBDA
    DO 6 I=1,NC
      J=NDIM-NC+I
  6 DERY(J)=RPHI(I)-CDE(I)+RAMBDA*G(I)
  1 RETURN
END

```

```

SUBROUTINE PRMAT(LC,U,V,R)
  DIMENSION U(7,7),V(7,7),R(7,7)
  DO 1 I=1,LC
    DO 1 J=1,NC
      R(I,J)=0.
    DO 1 K=1,NC
1  R(I,J)=R(I,J)+U(I,K)*V(K,J)
  RETURN
END

```

```

SUBROUTINE PROVEC(LC,U,V,R)
  DIMENSION U(7,7),V(7),R(7)
  DO 1 I=1,LC
    R(I)=0.
    DO 1 K=1,NC
1  R(I)=R(I)+U(I,K)*V(K)
  RETURN
END

```

```

SUBROUTINE COPY(X,Y,NCY,IGLF,NDLE,PRAT)
  DIMENSION Y(1),COPY(1),PRAT(5)
  RETURN
END

```


B

```

REAL K21,K22,K23,K24,K25,K51,K52,K53,K54,K55,MU,MUP,MU,NUP,I,J
LC=1
L=0
1 READ(LC,100) K21,K22,K23,K24,K25,K51,K52,K53,K54,K55
100 FORMAT (E12.5)
READ(LC,101) T
101 FORMAT (F5.1)
L=L+1

```

C DETECTION DE REGIME PERMANENT

```

A=1.224
B=0.10116
C=1.0
D=0.
E=0.1
F=0.80
G=0.8
H=68.0
I=1.07
J=32.
DELT2=30.
EPSI2=30.
DENOM=A*F+B*G
ALPHA=-B*D*I/DENOM
BETA=(B*E+B*F)/DENOM
GAMMA=-B*H/DENOM
ALPHP=+A*D*I/DENOM
BETAP=(B*G-A*E)/DENOM
GAMAP=A*H/DENOM

```

D CALCUL DES COEFFS DU VECTEUR PHI

```

MU=ALPHA*K22+ALPHP*(K24*F+K25)+I*K21
MUP=ALPHA*K52+ALPHP*(K54*F+K55)+I*K51
NU=BETA*K22+BETAP*(K24*F+K25)
NUP=BETA*K52+BETAP*(K54*F+K55)
RO=GAMMA*K22+GAMAP*(K24*F+K25)-J*K24
ROP=GAMMA*K52+GAMAP*(K54*F+K55)-J*K54

```

C EQUATIONS DES COMMANDES CALIBREES

C VALEUR D'AFFICHAGE DES POTENTIOMETRES

```

POT10=K21/(2.*DELT2)
POT11=20.*K22/DELT2
POT12=10.*K23/DELT2
POT47=20.*K24/DELT2
POT13=20.*K25/DELT2
POT14=-(MU+RO+1000.*NU)/(10.*DELT2)
POT42=-200.*(0.002*MU+NU)/DELT2
POT25=K51/(2.*EPSI2)
POT26=20.*K52/EPSI2
POT27=10.*K53/EPSI2
POT48=20.*K54/EPSI2
POT28=20.*K55/EPSI2
POT29=-(MUP+ROP+1000.*NUP)/(10.*EPSI2)
POT41=-200.*(0.002*MUP+NUP)/EPSI2
WRITE(3,199)

```

```

199 FORMAT ('O',20X,'EQUATIONS MACHINE CALIBREES')

```

```

WRITE(3,198) T

```

```

198 FORMAT (' ',F5.1)

```

```

199 FORMAT (' ',20X) ALPHA,BETA,GAMMA,ALPHP,BETAP,GAMAP

```

```

WRITE(3,202) MU,NU,RO,MUP,NUP,ROP
202 FORMAT (' ',2X,6(F10.5,5X))
SA=MU+RO
SC=NU+0.002*MU
SAP=MUP+ROP
SCP=NUP+0.002*MUP
WRITE(3,203)
203 FORMAT ('0',10X,'VECTEUR PHI LINEARISE')
WRITE(3,204) SA,NU,SC
204 FORMAT ('0',2X,'PHI1= ',F8.5,2X,F8.5,'QS ',F8.5,'DQS ')
WRITE(3,206) SAP,NUP,SCP
206 FORMAT ('0',2X,'PHI2= ',F8.3,2X,F8.5,'QS ',F8.5,'DQS ')
QS=1000.
DQS=0.
PHI1=SA+NU*QS+SC*DQS
PHI2=SAP+NUP*QS+SCP*DQS
WRITE(3,205) PHI1,PHI2
205 FORMAT ('0',2X,'PHI1 = ',F8.3,10X,'PHI2 = ',F8.3)
WRITE(3,200) POT10,POT11,POT12,POT47,POT13,POT14,POT42
200 FORMAT ('0',2X,'(U1/10)=',F9.5,'(P/5)',F8.5,'(QC/200)',F8.5,'(TETA
1/100)',F8.5,'(ZE/200)',F9.5,'(QD/200)',F8.5,2X,F8.5,'(DQS/2000)')
WRITE(3,201) POT25,POT26,POT27,POT48,POT28,POT29,POT41
201 FORMAT (' ',2X,'(U2/10)=',F9.5,'(P/5)',F8.5,'(QC/200)',F8.5,'(TETA
1/100)',F8.5,'(ZE/200)',F9.5,'(QD/200)',F8.5,2X,F8.5,'(DQS/2000)')
IF(L-1) 1,10,10
10 STOP
END

```


16/07/70

FORTMAIN

```
1DE+QK02(5)*X(4)+1.07*QK02(1)+0.00214*QK02(1)*BRUIT(M)-PHI1)
U(2)=(1./DELTA)*(QK05(1)*X(1)+QK05(2)*X(2)+QK05(3)*X(3)+QK05(4)*Z=
1DE+QK05(5)*X(4)+1.07*QK05(1)+0.00214*QK05(1)*BRUIT(M)-PHI2)
```

CALL A001

CALL A002

10 CONTINUE

GO TO 1

0 DD 11 I=1,6

DD 11 M=1,5

DD 11 J=1,8

DD 11 L=1,4

DD 11 K=1,4

A=BA(I)

B=TE(J)

C=FU(K)

D=FA(L)

E=BR(M)

C CALCUL DES COMMANDES U1 ET U2

X(1)=BAR(I)

X(2)=FUFL(K)

X(3)=TETA(J)

X(4)=FAU(L)

PHI1=32.51008-128.97-0.15545*BRUIT(M)

PHI2=-92.468+63.30+0.02462*BRUIT(M)

ZEDF=0.8*X(4)-32.

U(1)=(1./DELTA)*(QK02(1)*X(1)+QK02(2)*X(2)+QK02(3)*X(3)+QK02(4)*Z=

1DE+QK02(5)*X(4)+1.07*QK02(1)+0.00214*QK02(1)*BRUIT(M)-PHI1)

U(2)=(1./DELTA)*(QK05(1)*X(1)+QK05(2)*X(2)+QK05(3)*X(3)+QK05(4)*Z=

1DE+QK05(5)*X(4)+1.07*QK05(1)+0.00214*QK05(1)*BRUIT(M)-PHI2)

CALL A001

CALL A002

11 CONTINUE

KOKO=0

GO TO 1

7 LQLO=LQLO+1

GO TO(2,3),LQLO

2 DD 12 J=1,8

DD 12 M=1,5

DD 12 K=1,4

DD 12 L=1,4

DD 12 I=1,6

A=BA(I)

B=TE(J)

C=FU(K)

D=FA(L)

E=BR(M)

C CALCUL DES COMMANDES U1 ET U2

X(1)=BAR(I)

X(2)=FUFL(K)

X(3)=TETA(J)

X(4)=FAU(L)

PHI1=32.51008-128.97-0.15545*BRUIT(M)

PHI2=-92.468+63.30+0.02462*BRUIT(M)

ZEDF=0.8*X(4)-32.

16/07/70

FORTMAIN

107

```

      U(1)=(1./DELTA)*(QK02(1)*X(1)+QK02(2)*X(2)+QK02(3)*X(3)+QK02(4)*ZE
1DE+QK02(5)*X(4)+1.07*QK02(1)+0.00214*QK02(1)*BRUIT(M)-P(1))
      U(2)=(1./DELTA)*(QK05(1)*X(1)+QK05(2)*X(2)+QK05(3)*X(3)+QK05(4)*X(4)
1DE+QK05(5)*X(4)+1.07*QK05(1)+0.00214*QK05(1)*BRUIT(M)-P(2))

```

CALL A001

CALL A002

12 CONTINUE

JOJO=0

GO TO 1

3 DO 13 J=1,8

DO 13 L=1,4

DO 13 I=1,6

DO 13 M=1,5

DO 13 K=1,4

A=BA(I)

B=TE(J)

C=FU(K)

D=FA(L)

E=BR(M)

C- CALCUL DES COMMANDES U1 ET U2

X(1)=BAR(I)

X(2)=FUFL(X)

X(3)=TETA(J)

X(4)=FAU(L)

PHI1=32.51008-128.97-0.15545*BRUIT(M)

PHI2=-92.468+63.30+0.02462*BRUIT(M)

ZEDE=0.8*X(4)-32.

```

      U(1)=(1./DELTA)*(QK02(1)*X(1)+QK02(2)*X(2)+QK02(3)*X(3)+QK02(4)*ZE
1DE+QK02(5)*X(4)+1.07*QK02(1)+0.00214*QK02(1)*BRUIT(M)-P(1))

```

```

      U(2)=(1./DELTA)*(QK05(1)*X(1)+QK05(2)*X(2)+QK05(3)*X(3)+QK05(4)*X(4)
1DE+QK05(5)*X(4)+1.07*QK05(1)+0.00214*QK05(1)*BRUIT(M)-P(2))

```

CALL A001

CALL A002

13 CONTINUE

JOJO=0

LBLO=0

C-TEST DE CLASSEMENT DES ECHANTILLONS

1 DO 80 I=1,6

DO 80 J=1,8

DO 80 K=1,4

DO 80 L=1,4

DO 80 M=1,5

A=BA(I)

B=TE(J)

C=FU(K)

D=FA(L)

E=BR(M)

X(1)=BAR(I)

X(2)=FUFL(X)

X(3)=TETA(J)

X(4)=FAU(L)

PHI1=32.51008-128.97-0.15545*BRUIT(M)

PHI2=-92.468+63.30+0.02462*BRUIT(M)

ZEDE=0.8*X(4)-32.

16/07/70

FORTMAIN

0004

```

      U(1)=(1./DELTA)*(QK02(1)*X(1)+QK02(2)*X(2)+QK02(3)*X(3)+QK02(4)*ZF
1DE+QK02(5)*X(4)+1.07*QK02(1)+0.00214*QK02(1)*BRUIT(M)-PHI1)
      U(2)=(1./DELTA)*(QK05(1)*X(1)+QK05(2)*X(2)+QK05(3)*X(3)+QK05(4)*ZF
1DE+QK05(5)*X(4)+1.07*QK05(1)+0.00214*QK05(1)*BRUIT(M)-PHI2)
      CALL A001
      FA1=(WA1(A)+WA1(B)+WA1(C)+WA1(D)+WA1(E))*UA1
      IF(FA1) 82,81,81
81  V=V+1.
82  FA2=(WA2(A)+WA2(B)+WA2(C)+WA2(D)+WA2(E))*UA2
      IF(FA2) 84,83,83
83  W=W+1.
84  FB1=(WB1(A)+WB1(B)+WB1(C)+WB1(D)+WB1(E))*UB1
      IF(FB1) 86,85,85
85  Q=Q+1.
86  FB2=(WB2(A)+WB2(B)+WB2(C)+WB2(D)+WB2(E))*UB2
      IF(FB2) 88,87,87
87  Y=Y+1.
88  CONTINUE
      P1=V/T
      P2=Q/T
      P3=W/T
      P4=Y/T
88  WRITE(3,103)  P1,V,P2,Q,P3,W,P4,Y,Z
103  FORMAT (' ',2X,'P1=',F6.3,2X,'V=',F8.1,4X,'P2=',F6.3,2X,'Q=',F8.1,
14X,'P3=',F6.3,2X,'W=',F8.1,4X,'P4=',F6.3,2X,'Y=',F8.1,'Z=',I12)
      P5=P1+P2+P3+P4
      IF(P5-3.920) 98,89,89
89  DO 90 I=1,25
      WRITE(3,104)  I,WA1(I),I,WB1(I),I,WA2(I),I,WB2(I)
104  FORMAT ('C',2X,'WA1(',I2,')=',F7.3,8X,'WB1(',I2,')=',F7.3,8X,'WA2(
1',I2,')=',F7.3,8X,'WB2(',I2,')=',F7.3)
90  CONTINUE
      IF(P5-4) 98,99,99
98  GO TO 5
99  STOP
      END

```

DISK OPERATING SYSTEM/360 FORTRAN 360N-FD-451 CL 3-5

SUBROUTINE A001

INTEGER A,B,C,D,E,Z

COMMON Z,A,B,C,D,E,J(2),WA1(25),WA2(25),WB1(25),WB2(25),UA1,UA2,UB
11,UR2

QUANTIFICATION DES COMMANDES EN UA ET UB

IF(U(1)+0.200) 56,57,57

56 UA1=-1.

UB1=-1.

GO TO 60

57 IF(U(1)-0.200) 58,59,59

58 UA1=-1.

UB1=+1.

GO TO 60

59 UA1=+1.

UB1=+1.

60 IF(U(2)+0.025) 61,62,62

61 UA2=-1.

UB2=-1.

GO TO 65

62 IF(U(2)-0.025) 63,64,64

63 UA2=-1.

UB2=+1.

GO TO 65

64 UA2=+1.

UB2=+1.

65 RETURN

END

DISK OPERATING SYSTEM/360 FORTRAN 360N-ED-451 CL 3-

SUBROUTINE A002

INTEGER A,B,C,D,E,Z

COMMON Z,A,B,C,D,E,U(2),WA1(25),WA2(25),WB1(25),WB2(25),JA1,UA2,JB
11,UB2

C CALCUL DES VECTEURS DE PONDERATION

110 FA1=(WA1(A)+WA1(B)+WA1(C)+WA1(D)+WA1(E))*UA1

IF(FA1) 66,67,67

66 WA1(A)=WA1(A)+0.100*(1.-FA1)*UA1

WA1(B)=WA1(B)+0.100*(1.-FA1)*UA1

WA1(C)=WA1(C)+0.100*(1.-FA1)*UA1

WA1(D)=WA1(D)+0.100*(1.-FA1)*UA1

WA1(E)=WA1(E)+0.100*(1.-FA1)*UA1

Z=Z+1

GO TO 110

67 FA2=(WA2(A)+WA2(B)+WA2(C)+WA2(D)+WA2(E))*UA2

IF(FA2) 68,69,69

68 WA2(A)=WA2(A)+0.100*(1.-FA2)*JA2

WA2(B)=WA2(B)+0.100*(1.-FA2)*JA2

WA2(C)=WA2(C)+0.100*(1.-FA2)*JA2

WA2(D)=WA2(D)+0.100*(1.-FA2)*JA2

WA2(E)=WA2(E)+0.100*(1.-FA2)*JA2

Z=Z+1

GO TO 67

69 FB1=(WB1(A)+WB1(B)+WB1(C)+WB1(D)+WB1(E))*UB1

IF(FB1) 70,71,71

70 WB1(A)=WB1(A)+0.100*(1.-FB1)*UB1

WB1(B)=WB1(B)+0.100*(1.-FB1)*UB1

WB1(C)=WB1(C)+0.100*(1.-FB1)*UB1

WB1(D)=WB1(D)+0.100*(1.-FB1)*UB1

WB1(E)=WB1(E)+0.100*(1.-FB1)*UB1

Z=Z+1

GO TO 69

71 FB2=(WB2(A)+WB2(B)+WB2(C)+WB2(D)+WB2(E))*UB2

IF(FB2) 72,73,73

72 WB2(A)=WB2(A)+0.100*(1.-FB2)*JB2

WB2(B)=WB2(B)+0.100*(1.-FB2)*JB2

WB2(C)=WB2(C)+0.100*(1.-FB2)*JB2

WB2(D)=WB2(D)+0.100*(1.-FB2)*JB2

WB2(E)=WB2(E)+0.100*(1.-FB2)*JB2

Z=Z+1

GO TO 71

73 RETURN

END

D1

```

C      SUBOPTIMAL PROBLEM "JULY 70" 2 STATE VARIABLES, 2 CONSTRAINTS"
C      X2
C      ENTER WA AND WB
      DIMENSION WA(24), WB(24)
      WRITE(1,101)
101    FORMAT(6HWA(01),4X,6HWR(01))
      READ(0,001) (WA(01),WB(01))
201    FORMAT(2(F6.3,4X))
      WRITE(1,102)
102    FORMAT(6HWA(02),4X,6HWR(02))
      READ(0,201) (WA(02),WB(02))
      WRITE(1,103)
103    FORMAT(6HWA(03),4X,6HWR(03))
      READ(0,201) (WA(03),WB(03))
      WRITE(1,104)
104    FORMAT(6HWA(04),4X,6HWR(04))
      READ(0,201) (WA(04),WB(04))
      WRITE(1,105)
105    FORMAT(6HWA(05),4X,6HWR(05))
      READ(0,201) (WA(05),WB(05))
      WRITE(1,106)
106    FORMAT(6HWA(06),4X,6HWR(06))
      READ(0,201) (WA(06),WB(06))
      WRITE(1,107)
107    FORMAT(6HWA(07),4X,6HWR(07))
      READ(0,201) (WA(07),WB(07))
      WRITE(1,108)
108    FORMAT(6HWA(08),4X,6HWR(08))
      READ(0,201) (WA(08),WB(08))
      WRITE(1,109)
109    FORMAT(6HWA(09),4X,6HWR(09))
      READ(0,201) (WA(09),WB(09))
      WRITE(1,110)
110    FORMAT(6HWA(10),4X,6HWR(10))
      READ(0,201) (WA(10),WB(10))
      WRITE(1,111)
111    FORMAT(6HWA(11),4X,6HWR(11))
      READ(0,201) (WA(11),WB(11))
      WRITE(1,112)
112    FORMAT(6HWA(12),4X,6HWR(12))
      READ(0,201) (WA(12),WB(12))
      WRITE(1,113)
113    FORMAT(6HWA(13),4X,6HWR(13))
      READ(0,201) (WA(13),WB(13))
      WRITE(1,114)
114    FORMAT(6HWA(14),4X,6HWR(14))
      READ(0,201) (WA(14),WB(14))
      WRITE(1,115)
115    FORMAT(6HWA(15),4X,6HWR(15))
      READ(0,201) (WA(15),WB(15))
      WRITE(1,116)
116    FORMAT(6HWA(16),4X,6HWR(16))
      READ(0,201) (WA(16),WB(16))
      WRITE(1,117)
117    FORMAT(6HWA(17),4X,6HWR(17))
      READ(0,201) (WA(17),WB(17))
      WRITE(1,118)
118    FORMAT(6HWA(18),4X,6HWR(18))
      READ(0,201) (WA(18),WB(18))
      WRITE(1,119)
119    FORMAT(6HWA(19),4X,6HWR(19))
      READ(0,201) (WA(19),WB(19))
      WRITE(1,120)
120    FORMAT(6HWA(20),4X,6HWR(20))
      READ(0,201) (WA(20),WB(20))
      WRITE(1,121)
121    FORMAT(6HWA(21),4X,6HWR(21))
      READ(0,201) (WA(21),WB(21))

P121    FORMAT(6H122)
122    FORMAT(6HWA(22),4X,6HWR(22))
      READ(0,201) (WA(22),WB(22))
      WRITE(1,123)
123    FORMAT(6HWA(23),4X,6HWR(23))
      READ(0,201) (WA(23),WB(23))
      WRITE(1,124)
124    FORMAT(6HWA(24),4X,6HWR(24))
      READ(0,201) (WA(24),WB(24))
      WRITE(5,202) (WA(I),WB(I),I=1,24)
202    FORMAT(2F6.3)
      STOP
      END

```

```

C      SUBOPTIMAL PROBLEM "MAY 70" WITH 5 STATE VARIABLES
C      ENTREE DE WAI WBI WAP WR2
      DIMENSION WAI(25), WBI(25), WAP(25), WR2(25)
      WRITE(1,101)
101    FORMAT(7HWA1(01),4X,7HWBI(01),4X,7HWA2(01),4X,7HWR2(01))
      READ(0,201) (WAI(01),WBI(01),WAP(01),WR2(01))
201    FORMAT(4F6.3,5X)
      WRITE(1,102)
102    FORMAT(7HWA1(02),4X,7HWBI(02),4X,7HWA2(02),4X,7HWR2(02))
      READ(0,201) (WAI(02),WBI(02),WAP(02),WR2(02))
      WRITE(1,103)
103    FORMAT(7HWA1(03),4X,7HWBI(03),4X,7HWA2(03),4X,7HWR2(03))
      READ(0,201) (WAI(03),WBI(03),WAP(03),WR2(03))
      WRITE(1,104)
104    FORMAT(7HWA1(04),4X,7HWBI(04),4X,7HWA2(04),4X,7HWR2(04))
      READ(0,201) (WAI(04),WBI(04),WAP(04),WR2(04))
      WRITE(1,105)
105    FORMAT(7HWA1(05),4X,7HWBI(05),4X,7HWA2(05),4X,7HWR2(05))
      READ(0,201) (WAI(05),WBI(05),WAP(05),WR2(05))
      WRITE(1,106)
106    FORMAT(7HWA1(06),4X,7HWBI(06),4X,7HWA2(06),4X,7HWR2(06))
      READ(0,201) (WAI(06),WBI(06),WAP(06),WR2(06))
      WRITE(1,107)
107    FORMAT(7HWA1(07),4X,7HWBI(07),4X,7HWA2(07),4X,7HWR2(07))
      READ(0,201) (WAI(07),WBI(07),WAP(07),WR2(07))
      WRITE(1,108)
108    FORMAT(7HWA1(08),4X,7HWBI(08),4X,7HWA2(08),4X,7HWR2(08))
      READ(0,201) (WAI(08),WBI(08),WAP(08),WR2(08))
      WRITE(1,109)
109    FORMAT(7HWA1(09),4X,7HWBI(09),4X,7HWA2(09),4X,7HWR2(09))
      READ(0,201) (WAI(09),WBI(09),WAP(09),WR2(09))
      WRITE(1,110)
110    FORMAT(7HWA1(10),4X,7HWBI(10),4X,7HWA2(10),4X,7HWR2(10))
      READ(0,201) (WAI(10),WBI(10),WAP(10),WR2(10))
      WRITE(1,111)
111    FORMAT(7HWA1(11),4X,7HWBI(11),4X,7HWA2(11),4X,7HWR2(11))
      READ(0,201) (WAI(11),WBI(11),WAP(11),WR2(11))
      WRITE(1,112)
112    FORMAT(7HWA1(12),4X,7HWBI(12),4X,7HWA2(12),4X,7HWR2(12))
      READ(0,201) (WAI(12),WBI(12),WAP(12),WR2(12))
      WRITE(1,113)
113    FORMAT(7HWA1(13),4X,7HWBI(13),4X,7HWA2(13),4X,7HWR2(13))
      READ(0,201) (WAI(13),WBI(13),WAP(13),WR2(13))
      WRITE(1,114)
114    FORMAT(7HWA1(14),4X,7HWBI(14),4X,7HWA2(14),4X,7HWR2(14))
      READ(0,201) (WAI(14),WBI(14),WAP(14),WR2(14))
      WRITE(1,115)
115    FORMAT(7HWA1(15),4X,7HWBI(15),4X,7HWA2(15),4X,7HWR2(15))
      READ(0,201) (WAI(15),WBI(15),WAP(15),WR2(15))
      WRITE(1,116)
116    FORMAT(7HWA1(16),4X,7HWBI(16),4X,7HWA2(16),4X,7HWR2(16))
      READ(0,201) (WAI(16),WBI(16),WAP(16),WR2(16))
      WRITE(1,117)
117    FORMAT(7HWA1(17),4X,7HWBI(17),4X,7HWA2(17),4X,7HWR2(17))
      READ(0,201) (WAI(17),WBI(17),WAP(17),WR2(17))
      WRITE(1,118)
118    FORMAT(7HWA1(18),4X,7HWBI(18),4X,7HWA2(18),4X,7HWR2(18))
      READ(0,201) (WAI(18),WBI(18),WAP(18),WR2(18))
      WRITE(1,119)
119    FORMAT(7HWA1(19),4X,7HWBI(19),4X,7HWA2(19),4X,7HWR2(19))
      READ(0,201) (WAI(19),WBI(19),WAP(19),WR2(19))
      WRITE(1,120)
120    FORMAT(7HWA1(20),4X,7HWBI(20),4X,7HWA2(20),4X,7HWR2(20))
      READ(0,201) (WAI(20),WBI(20),WAP(20),WR2(20))
      WRITE(1,121)
121    FORMAT(7HWA1(21),4X,7HWBI(21),4X,7HWA2(21),4X,7HWR2(21))
      READ(0,201) (WAI(21),WBI(21),WAP(21),WR2(21))
      WRITE(1,122)
122    FORMAT(7HWA1(22),4X,7HWBI(22),4X,7HWA2(22),4X,7HWR2(22))
      READ(0,201) (WAI(22),WBI(22),WAP(22),WR2(22))
      WRITE(1,123)
123    FORMAT(7HWA1(23),4X,7HWBI(23),4X,7HWA2(23),4X,7HWR2(23))
      READ(0,201) (WAI(23),WBI(23),WAP(23),WR2(23))
      WRITE(1,124)
124    FORMAT(7HWA1(24),4X,7HWBI(24),4X,7HWA2(24),4X,7HWR2(24))
      READ(0,201) (WAI(24),WBI(24),WAP(24),WR2(24))
      WRITE(1,125)
125    FORMAT(7HWA1(25),4X,7HWBI(25),4X,7HWA2(25),4X,7HWR2(25))
      READ(0,201) (WAI(25),WBI(25),WAP(25),WR2(25))
      WRITE(5,203) (WAI(1),WBI(1),WAP(1),WR2(1),I-1,25)
203    FORMAT(4F6.3)
      STOP
      END

```

E₁

```

C      TRAJECTOIRES SOUS OPTIMALES, PR EN 1/52 AVEC BUTEES SUR X2
C      CALCUL DE LA COMMANDE QUANTIFIEE
      INTEGER A,B
      LOGICAL SENSW
      DIMENSION WA(24),WB(24)
      DO 10 I=1,24
      READ(4,303)(WA(I),WB(I))
303    FORMAT(2F6.3)
10     CONTINUE
1      CALL QPRADR(X1,0,1,IFRA)
      X1=X1*5.
      CALL QRBADR(X2,1,1,IFBR)
      X2=X2*0.05
      CALL QRBADR(T,2,1,IFRC)
      T=T*222.
      TETA=200.-T
      IF(SENSW(1))GO TO 2
      GO TO 3
2      WRITE(3,100)(X1,X2,T,TETA)
100    FORMAT(F7.4,3X,F7.4,3X,F5.1,3X,F5.1)
C      QUANTIFICATION DES ETATS
3      IF(X1+1.5)15,16,16
15     A=1
      GO TO 41
16     IF(X1+1.25)17,18,18
17     A=2
      GO TO 41
18     IF(X1+1.)19,20,20
19     A=3
      GO TO 41
20     IF(X1+0.75)21,22,22
21     A=4
      GO TO 41
22     IF(X1+0.5)23,24,24
23     A=5
      GO TO 41
24     IF(X1+0.25)25,26,26
25     A=6
      GO TO 41
26     IF(X1)27,28,28
27     A=7
      GO TO 41
28     IF(X1-0.25)29,30,30
29     A=8
      GO TO 41
30     IF(X1-0.5)31,32,32
31     A=9
      GO TO 41
32     IF(X1-0.75)33,34,34
33     A=10
      GO TO 41
34     IF(X1-1.)35,36,36
35     A=11
      GO TO 41
36     IF(X1-1.25)37,38,38
37     A=12
      GO TO 41
38     IF(X1-1.5)39,40,40
39     A=13
      GO TO 41

```

```

42 A=14
43 IF(X2+0.00142,43,43
44 GO TO 60
45 IF(X2+0.0015)46,43
46 R=16
47 GO TO 60
48 IF(X2+0.00156,47,
49 GO TO 60
50 IF(X2+0.00158,49,49
51 R=18
52 GO TO 60
53 IF(X2)50,51,51
54 R=18
55 GO TO 60
56 IF(X2-0.00502,51,54
57 R=20
58 GO TO 60
59 IF(X2-0.010)54,55,55
60 R=21
61 GO TO 60
62 IF(X2-0.015)56,57,57
63 B=22
64 GO TO 60
65 IF(X2-0.020)58,59,59
66 R=23
67 GO TO 60
68 R=24
69 FA=WA(A)+WA(R)
70 FB=WB(A)+WB(R)
71 IF(SENSW(3))GO TO 9
72 GO TO 99
73 WRITE(3,500)(A,B)
74 FORMAT(2(I2,5X))
75 WRITE(3,501)(FA,FB)
76 FORMAT(2(F7.3,3X))
77 IF(FA)66,67,67
78 IIA=-1.
79 GO TO 68
80 IIA=+1.
81 IF(FB)69,70,70
82 UB=-1.
83 GO TO 71
84 IIB=+1.
85 IF(IIA+IIB)72,73,74
86 IIE=-1.
87 GO TO 99
88 IIE=0.
89 GO TO 99
90 IIE=+1.
91 IF(X2+0.020)91,92,92
92 IF(X1+1.5)95,96,92
93 IF(X2-0.020)75,76,94
94 IF(X1+1.5)95,96,75
95 IIE=0.
96 IF(SENSW(2))GO TO 4
97 GO TO 80
98 WRITE(3,600)(II)
99 FORMAT(1A1,85,2X)
100 CALL OVERDAR(II,0.1,1E-10)
101 IF(II-10.)1.1,76
102 STOP
103 END

```

E₂ - Quantification 1

```

C   TRAJECTOIRES SS OPTIMALES, CALCUL DES COMMANDES
    INTEGEP A,B,C,D,E
    LOGICALSENW
    DIMENSION WA1(25),WB1(25),WA2(25),WB2(25)
    DO 10 I=1,25
    READ(4,303)(WA1(I),WB1(I),WA2(I),WB2(I))
303  FORMAT(4F6.3)
10   CONTINUE
1    CALL QRBADR(BAR,0.1,IERA)
    BAR=BAR*5.
    CALL QRBADR(TETA,1.1,IFRR)
    TETA=TETA*100.
    CALL QRBADR(FUEL,2.1,IERC)
    FUEL=FUEL*200.
    CALL QRBADR(EAU,3.1,IFRD)
    EAU=EAU*200.
    CALL QRBADR(BRUIT,4.1,IFRF)
    BRUIT=BRUIT*2000.
    IF(SENW(1))GO TO2
    GO TO 3
2    WRITE(3,100)(BAR,TETA,FUEL,EAU,BRUIT)
100  FORMAT(F7.4,8Y,F7.2,2X,F7.2,2X,F7.2,8Y,F8.2)
C   QUANTIFICATION DES ETATS ECHANTILLONNES
3    IF(BAR+1.)15,16,16
15   A=1
    GO TO 25
16   IF(BAR+0.4)17,18,18
17   A=2
    GO TO 25
18   IF(BAR)19,20,20
19   A=3
    GO TO 25
20   IF(BAR-0.4)21,22,22
21   A=4
    GO TO 25
22   IF(BAR-1.)23,24,24
23   A=5
    GO TO 25
24   A=6
25   IF(TETA+0.8)26,27,27
26   B=7
    GO TO 34
27   IF(TETA+0.3)28,29,29
28   B=8
    GO TO 34
29   IF(TETA)400,401,401
400  B=9
    GO TO 34
401  IF(TETA-0.3)402,31,31
402  B=10
    GO TO 34
31   IF(TETA-0.8)32,33,33
32   B=11
    GO TO 34
33   B=12
34   IF(FUEL-59.)35,36,36
35   C=13
    GO TO 41
36   IF(FUEL-77.)37,38,38
37   C=14
    GO TO 41
38   IF(FUEL-95.)39,40,40
39   C=15
    GO TO41

```

```

40      C=16
41      IF(DAII-30.)42,43,43
42      D=17
         GOTO 48
43      IF(EAU-40.)44,45,45
44      D=18
         GO TO 48
45      IF(EAU-50.)46,47,47
46      D=19
         GO TO 48
47      D=20
48      IF(BRUIT+150.)49,50,50
49      E=21
         GO TO 55
50      IF(BRUIT+50.)51,52,52
51      E=22
         GO TO 55
52      IF(BRUIT-50.)53,54,54
53      E=23
         GO TO 55
54      IF(BRUIT-150.)403,404,404
403     F=24
         GO TO 55
404     E=25

55      FA1=WA1(A)+WA1(B)+WA1(C)+WA1(D)+WA1(E)
         FB1=WB1(A)+WB1(B)+WB1(C)+WB1(D)+WB1(E)
         FA2=WA2(A)+WA2(B)+WA2(C)+WA2(D)+WA2(E)
         FB2=WB2(A)+WB2(B)+WB2(C)+WB2(D)+WB2(E)
         IF(SENSW(3))GOTO 9
         GO TO 900
9        WRITE(3,500)(A,B,C,D,E)
500     FORMAT(5(I2,5X))
         WRITE(3,501)(FA1,FB1,FA2,FB2)
501     FORMAT(4(F7.3,2X))
90      IF(FA1)56,57,57
56      UA1=-1.
         GO TO 58
57      UA1=+1.
58      IF(FB1)59,60,60
59      UB1=-1.
         GO TO 61
60      UB1=+1.
61      IF(UA1+UB1)62,63,64
62      U1=-1.
         GO TO 65
63      U1=0.
         GO TO 65
64      U1=+1.
65      IF(FA2)66,67,67
66      UA2=-1.
         GO TO 68
67      UA2=+1.
68      IF(FB2)69,70,70
69      UB2=-1.
         GO TO 71
70      UB2=+1.
71      IF(UA2+UB2)72,73,74
72      U2=-1.
         GO TO 75
73      U2=0.
         GO TO 75
74      U2=+1.
75      IF(SENSW(2))GO TO 4
         GO TO 80
4        WRITE(3,600)(U1,U2)
600     FORMAT(F5.2,10X,F5.2)
80      CALL QWBDAR(U1,0,1,IFRF)
         CALL QWBDAR(U2,1,1,IFRG)
         IF(U1-10.)1,1,76
76      STOP
         END

```

E₂ - Quantification 2

```

C   TRAJECTOIRES SS OPTIMALES , CALCUL DES COMMANDES
    INTEGER A,R,C,D,E
    LOGICAL SENSW
    DIMENSION WA1(25),WR1(25),WA2(25),WR2(25)
    DO 10 I=1,25
303 READ(4,303)(WA1(I),WR1(I),WA2(I),WR2(I))
    FORMAT(4F6.3)
10  CONTINUE
1   CALL QPRADP(RAR,0.1,IFRA)
    RAR=BAR*5.
    CALL ORBADP(TETA,1.1,IFRR)
    TETA=TETA*10.
    CALL QPRADR(FUEL,2.1,IERC)
    FUEL=FUEL*200.
    CALL QPRADR(EAU,3.1,IFRD)
    EAU=EAU*200.
    CALL QPRADR(BRUIT,4.1,IFRE)
    BRUIT=BRUIT*200.
    IF(SENSW(1))GO TO 2
    GO TO 3
2   WRITE(3,100)(RAR,TETA,FUEL,EAU,BRUIT)
100 FORMAT(F7.4,8X,F7.2,8X,F7.2,8X,F7.2,8X,F8.2)
C   QUANTIFICATION DES ETATS ECHANTILLONNES
3   IF(BAR+1.)15,16,16
15  A=1
    GO TO 25
16  IF(BAR+0.4)17,18,18
17  A=2
    GO TO 25
18  IF(BAR)19,20,20
19  A=3
    GO TO 25
20  IF(BAR-0.4)21,22,22
21  A=4
    GO TO 25
22  IF(BAR-1.)23,24,24
23  A=5
    GO TO 25
24  A=6
25  IF(TETA+0.8)26,27,27
26  R=7
    GO TO 34
27  IF(TETA+0.3)28,29,29
28  B=8
    GO TO 34
29  IF(TETA)400,401,401
400 B=9
    GO TO 34
401 IF(TETA-0.3)402,31,31
402 R=10
    GO TO 34
31  IF(TETA-0.8)32,33,33
32  B=11
    GO TO 34
33  B=12
34  IF(FUEL-65.)35,36,36
35  C=13
    GO TO 41
36  IF(FUEL-80.)37,38,38
37  C=14
    GO TO 41
38  IF(FUEL-95.)39,40,40
39  C=15
    GO TO 41

```

```

40      C=16
41      IF(EAU-35.)42,43,43
42      D=17
         GO TO 48
43      IF(EAU-40.)44,45,45
44      D=18
         GO TO 48
45      IF(EAU-45.)46,47,47
46      D=19
         GO TO 48
47      D=20
48      IF(BRUIT+150.)49,50,50
49      E=21
         GO TO 55
50      IF(BRUIT+50.)51,52,52
51      E=22
         GO TO 55
52      IF(BRUIT-50.)53,54,54
53      E=23
         GO TO 55
54      IF(BRUIT-150.)403,404,404
403     E=24
         GO TO 55
404     E=25

55      FA1=WAI(C)+WAI(B)+WAI(C)+WAI(D)+WAI(E)
         FB1=WB1(A)+WB1(B)+WB1(C)+WB1(D)+WB1(F)
         FA2=WA2(A)+WA2(B)+WA2(C)+WA2(D)+WA2(F)
         FB2=WB2(A)+WB2(B)+WB2(C)+WB2(D)+WB2(F)
         IF(SENSW(3))GO TO 9
         GO TO 90
9        WRITE(3,500)(A,B,C,D,F)
500     FORMAT(5(I2,5X))
         WRITE(3,501)(FA1,FB1,FA2,FB2)
501     FORMAT(4(F7.3,2X))
90      IF(FA1)56,57,57
56      UA1=-1.
         GO TO 58
57      UA1=+1.
58      IF(FB1)59,60,60
59      UB1=-1.
         GO TO 61
60      UB1=+1.
61      IF(UA1+UB1)62,63,64
62      U1=-1.
         GO TO 65
63      U1=0.
         GO TO 65
64      U1=+1.
65      IF(FA2)66,67,67
66      UA2=-1.
         GO TO 68
67      UA2=+1.
68      IF(FB2)69,70,70
69      UB2=-1.
         GO TO 71
70      UB2=+1.
71      IF(UA2+UB2)72,73,74
72      U2=-1.
         GO TO 75
73      U2=0.
         GO TO 75
74      U2=+1.
75      IF(SENSW(2))GO TO 4
         GO TO 80
4        WRITE(3,600)(U1,U2)
600     FORMAT(F5.2,10X,F5.2)
80      CALL QWRDAR(U1,0,1,IFR1)
         CALL QWRDAR(U2,1,1,IFR2)
         IF(U1-10.)1,1,76
76      STOP
         END

```